A TOPOLOGICAL THEORY OF WEAVING AND ITS APPLICATIONS IN COMPUTER GRAPHICS

A Dissertation

by

SHIYU HU

Submitted to the Office of Graduate and Professional Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Chair of Committee,	Jianer Chen
Committee Members,	Ergun Akleman
	Andreas Klappenecker
	Scott Schaefer
Head of Department,	Duncan Walker

August 2013

Major Subject: Computer Science

ABSTRACT

Recent advances in the computer graphics of woven images on surfaces in 3-space motivate the development of weavings for arbitrary genus surfaces. We present herein a general framework for weaving structures on general surfaces in 3-space, and through it, we demonstrate how weavings on such surfaces are inducible from connected graph imbeddings on the same surfaces. The necessary and sufficient conditions to identify the inducible weavings in our framework are also given. For low genus surfaces, like plane and torus, we extend our framework to the weavings which are inducible from disconnected imbedded graphs. In particular, we show all weavings on a plane are inducible in our framework, including most Celtic Knots.

Moreover, we study different weaving structures on general surfaces in 3-space based on our framework. We show that any weaving inducible in our framework can be converted into an alternating weaving by appropriately changing the strand orders at some crossings. By applying a topological surgery operation, called doubling operation, we can refine a weaving or convert certain non-twillable weavings into twillable weavings on the same surfaces. Interestingly, two important subdivision algorithms on graphs imbeddings, the Catmull-Clark and Doo-Sabin algorithms, correspond nicely to our doubling operation on induced weavings. Another technique we used in studying weaving structures is repetitive patterns. A weaving that can be converted into a twillable weaving by our doubling operation has a highly-symmetric structure, which consists of only two repetitive patterns. An extension of the symmetric structure leads to Quad-Pattern Coverable meshes, which can be seamlessly covered with only one periodic pattern. Both of these two topological structures can be represented with simple Permutation Voltage graphs.

A considerable advantage of our model is that it is topological. This permits the graphic designer to superimpose strand colors and geometric attributes — distances, angles, and curvatures — that conform to manufacturing or artistic criteria.

We also give a software example for plane weaving construction. A benefit of the software is that it supports plane weaving reconstructions from an image of a plane weaving, which could be useful for recording and modifying existing weavings in real life.

DEDICATION

To my family and friends

ACKNOWLEDGEMENTS

First, I would like to express the deepest appreciation to my advisor, Professor Jianer Chen, for his patience, motivation, immerse knowledge and continuous support throughout the course of this research. His comments can always inspire me and give me a broader view of the problems. Without his kind and unreserved help, it is impossible for me to finish this thesis.

My sincere thanks also go to Professor Ergun Akleman for the constructive advices and encouragements in the research of weaving applications in computer graphics. I also want to extend my gratitude to Professor Jonathan L. Gross for the guidance in formal mathematical thinking and expressions. Moreover, I appreciate the rest of my committee members, Professor Andreas Klappenecker, and Professor Scott Schaefer, for their valuable suggestions in my proposal and final defences.

I would also like to thank my fellow labmates: Qing Xing, Yixin Cao, Cheng Cao, Jia-Hao Fan, Wenjun Li, Youyou Wang and Pradeep Garigipati for the helpful discussions and the great study time we experienced together at Texas A&M University.

Finally, thanks to my parents, Peixin Hu and Meiying Shen. Their love and tireless dedications are my greatest motivator.

TABLE OF CONTENTS

Page

ABSTRACT ii

DEDICATION iv

ACKNOWLEDGEMENTS v

TABLE OF CONTENTS vi

LIST OF FIGURES ix

1. INTRODUCTION 1

2. LITERATURE REVIEW 7

2.1 Weaving Construction 7
2.2 Edge Twisting 8
2.3 Twillable Weaving 9
2.4 Texture Mapping 10
2.5 Permutation Voltage Graph 10

3. PRELIMINARY 12

3.1 2-Manifolds and Surfaces 12
3.2 Graph Imbeddings and Graph Rotation Systems 13
3.3 Weaving on Topological Surfaces 15
3.4 The Extended Edge-Twisting Operation 16

4. CELLULAR WEAVING AND NORMAL WEAVING 21

4.1 Cellular Weavings 21
4.1.1 Medial graphs 23
4.1.2 Topological edge-contraction 25
4.2 Normal Weaving 27

5. WEAVINGS ON SPHERE AND TORUS 35

5.1 Plane Weaving 35
5.1.1 Weaving from plane graph 36
5.1.2 Plane graph from plane weaving 38
5.2 Weaving on Torus 45

6. WEAVING DOUBLING OPERATIONS AND IMBEDDINGS SUBDIVISION OPERATIONS . . . 52
6.1 Catmull-Clark Subdivisions . . . 52
6.2 The Doubling Operation on Weavings . . . 53
6.3 Doo-Sabin Subdivisions . . . 58
6.4 Dual Imbeddings . . . 59
7. ALTERNATING WEAVINGS, TWILLABLE WEAVINGS AND CHILD-TWILLABLE WEAVINGS . . . 62
7.1 Alternating Weaving and Twillable Weaving . . . 62
7.2 Child-Twillable Weaving . . . 66
7.2.1 Identification of child-twillable weavings . . . 67
7.2.2 Descendants of child-twillable weavings . . . 80
8. QUAD-PATTERN COVERABLE MESHES . . . 84
8.1 Identification of QPC Meshes . . . 85
8.2 QPC-Preserving Valence Reduction . . . 91
8.3 Non-Separating Cycles in Dual of QPC Meshes . . . 95
9. PERMUTATION VOLTAGE GRAPHS FOR CHILD-TWILLABLE WEAVING AND QUAD-PATTERN COVERABLE MESHES . . . 97
9.1 Permutation Voltage Graph for Child-Twillable Weaving . . . 98
9.1.1 Construct child-twillable weaving . . . 99
9.1.2 Child-twillable mesh on any positive genus . . . 102
9.2 Permutation Voltage Graph for QPC Meshes . . . 104
9.2.1 Construct QPC meshes . . . 104
9.2.2 QPC meshes on any positive genus surface . . . 108
10. SOFTWARE DESIGN: PLANE WEAVING CONSTRUCTION AND RECONSTRUCTION . . . 111
10.1 Overall Description . . . 111
10.2 Planar Graph Drawing and Induced Plane Weaving . . . 113
10.2.1 Insert vertex and edge . . . 113
10.2.2 Vertex editing . . . 114
10.2.3 Edge editing . . . 115
10.2.4 Plane weaving construction . . . 117
10.2.5 Adjusting the width of strands . . . 119
10.2.6 Relocating control points . . . 119
10.2.7 Changing twisted types . . . 119
10.3 Weaving Recognition and Reconstruction . . . 120
10.3.1 Image binarization . . . 120
10.3.2 Contour detection . . . 121
10.3.3 Gaps coloring . . . 121
10.3.4 Graph construction . . . 121

11. CONCLUSION 134

REFERENCES 139

LIST OF FIGURES

FIGURE		Page
1.1	Examples of plain, twill and satin weaving	1
1.2	Examples of Celtic knots	2
1.3	Twisting the two sides of an edge	3
1.4	The correspondence between an imbedded graph and the induced weaving.	4
1.5	Examples of weavings on orientable surfaces	4
2.1	The boundary walk of paper-strip sculptures are links[2]	8
2.2	Diagonal twill pattern on weaving[1]	9
3.1	(a) an untwisted edge. (b) a counterclockwise twisted edge. (c) a clockwise twisted edge. (d) a double-counterclockwise twisted edge. (e) a double-clockwise twisted edge.	17
4.1	(a) The graph K_4 with an ERS and the induced weaving. (b) A multi-color medial graph for the imbedding $K_4 \to S_0$.	24
4.2	Examples of non-cellular weaving patterns (from [49])	28
4.3	(a) Incipient induced weaving for an ERS, after Steps (S0), (S1), and (S2). (b) Induced weaving, after Steps (S3) and (S4).	30
4.4	Two representations of a gap of genus 2 with two holes. (a) An 8-sided polygon with two (red) holes. (b) The corresponding imbedding of the bouquet B_5 in a two-holed surface of genus 2.	31
4.5	(a) Virtual merging of a 3-sided hole and a 4-sided hole into a 7-sided hole. (b) Seven "semi-edges" from the corner c to the seven corners of the polygonal hole, and the seven resulting "semi-faces".	32
4.6	Twisting a connecting edge (green) to be consistent with the given weaving (red).	33
5.1	An example of the tree of a plane graph	37

5.2 F-plane graph and V-plane graph based on the same weaving 39
5.3 Algorithm to construct F-planar graph 40
5.4 The gap q corresponds to a region of $\iota(G_*^{k-1})$ 42
5.5 The gap q corresponds to a vertex of $\iota(G_*^{k-1})$ 43
5.6 Algorithm to construct general planar graph 44
5.7 If the two gaps q_1, q_2 correspond to two vertices in $\iota(G_*^k)$ 49
6.1 Catmull-Clark subdivision 53
6.2 A weaving, doubled and redoubled. 54
6.3 2-coloring the gaps of a doubled weaving. 55
6.4 (a) The weaving strands induced by a simply-twisted ERS. (b) The Catmull-Clark subdivision of the imbedded graph that induces the weaving. (c) Doubling the strands. 56
6.5 Doo-Sabin subdivision of a graph imbedding $G \to S_i$ 59
6.6 (a) A graph imbedding. (b) Superposition of the Clark-Catmull (blue) and Doo-Sabin (red) subdivisions. 60
7.1 Examples of weavings on orientable surfaces 62
7.2 The alternating weaving induced by a uni-direction-twisted rotation system. 64
7.3 "#" structures after doubling strands 67
7.4 Two basic feasible "#" structures 67
7.5 Corner costs at a cross 70
7.6 The corner precedence graph (CPG) D_0 72
7.7 The algorithm *Placement* 77
7.8 Descendants of child-twillable structures 82
8.1 The quadrangular meshes covered with a periodic quad-pattern[26] . 84
8.2 The corner precedence graph (CPG) D_1 87
8.3 Reducing face-size 92

8.4 Reduction of a 16-valent vertex into three 8-valent vertices, while preserving QPC property.[26] 94

9.1 An imbedded base graph D_4 99

9.2 A imbedded bouquet B_2 with two self-circles 105

10.1 Main panel of WvConstructor 112

10.2 Insert vertices and edges (vertex with green color is an active vertex) (a1-b1) Insert a new vertex; (a2-b2) Insert a new vertex and a new edge between the new vertex and active vertex; (a3-b3) Insert a new edge between active vertex and an existing vertex. 114

10.3 Vertex Relocation and Deletion. (a1-b1)Draw a vertex to a new position; (a2-b2) Delete the chosen vertex 115

10.4 Edge Contraction, Edge Deletion and Changing Twist-type. (a1-b1)Contract an edge; (a2-b2) Delete an edge; (a3-d3) Changing the twisted-type of the bottom edge. 116

10.5 Five control points attached to each edge 117

10.6 A weaving curve defined by four control points $\{c_1, c_5, c_2', c_1'\}$ for face corner (v, e_1, e_2). 118

10.7 A sequence of drawing operations and their updated induced weaving. (1) create a single vertex a; (2) create a new vertex b and a new edge (a, b); (3) create a new vertex c and a new edge (b, c); (4) create a new vertex d and a new edge (d, c); (5) drew vertex d to a new position; (6) create a new edge (d, b); (7) delete edge (b, c); (8) contract edge (d, c) into a vertex $c'(d')$; (9) delete vertex $c'(d')$ . . . 123

10.8 Adjusting the width of strands 123

10.9 A sequence of control points relocation operations and the appearance of weaving. (1) Control points in default positions; (2) Relocate c_3, the position of c_5 is also changed; (3) Relocate c_1, all the other four control points are also relocated. 124

10.10 The bottom edge (v_1, v_2) with four types of twisted-types in WvConstructor, 1^+-twisted, 1^--twisted, and two types of NULL-twisted 124

10.11 The pipeline of Weaving Reconstruction 124

10.12 An example for image binarization. (a) Input image; (b) Binary image125

10.13 An example of weaving contours detection. (a) The detected contours; (b) Each contour defines a gap, p_0 is the unbounded exterior gap. 125

10.14 Algorithm for classify gaps into F-gap and V-gap 126

10.15 An example of finding adjacent contours. The small cycles are the centers of edges of the contours. The short segments incident to the small cycles are rays to find adjacent contours. 126

10.16 Algorithm for classify gaps into F-gap and V-gap 127

10.17 An example of graph construction(a) and its induced weaving(b). . 128

10.18 An example for weaving reconstruction(1). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving 129

10.19 An example for weaving reconstruction(2). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving 130

10.20 An example for weaving reconstruction(3). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving 131

10.21 An example for Trefoil knot reconstruction(4). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving 132

10.22 An example for weaving reconstruction(5). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving 133

1. INTRODUCTION

Weaving, as a kind of traditional fabric productions, has existed for thousands of years. Of importance in human history and society, clothes, baskets and sweaters are examples of weaving productions. In traditional weaving structures, there are two types of threads, *warp* and *weft*. A warp thread is longitudinal and a weft thread is lateral. These two types of threads go over or under each other at *crosses*. The different orders of warp and weft threads at crosses in a weaving lead to different appearances.

Most of the weaving patterns can be constructed with three fundamental weaving patterns, *plain weaving*, *twill weaving* and *satin weaving* [14], as shown in Figure 1.1. In a plain weaving, each weft thread repeatedly goes over and then under a warp thread, and its neighbor weft thread goes under and then over repeatedly. In a classical twill weaving, a weft thread goes over two warp threads and then under two warp threads, and two neighbor weft threads have one step offset which provides a beautiful diagonal pattern. The most important feature of a satin weaving is its lustrous appearance, which comes from its thread structure: a weft thread repeatedly goes under four or more warp threads and goes over only one. This structure makes light reflection consistent in most areas of a satin weaving.

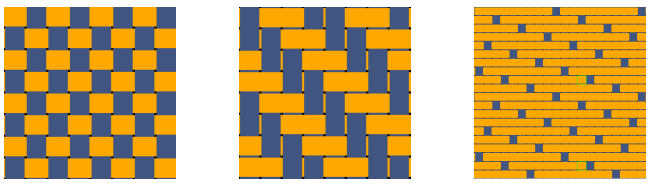


Figure 1.1: Examples of plain, twill and satin weaving

In particular, *Celtic knots*, as an artwork originating at the late Roman Empire [48], is a variety of weave work in the plane. The interlace structures of Celtic knots, as shown in Figure 1.2, are widely used in decoration and ornamentation, which attract great interests from artists.

Figure 1.2: Examples of Celtic knots

A recently developed method for modeling weavings on an arbitrary topological surface is based on *extended graph rotation systems* [1, 3]. If we imagine each face-boundary walk (abbr. ***fb-walk***) of a graph imbedding $\pi : G \to S$ to be lying slightly inside the face, the fb-walks form a collection of disjoint cycles on the orientable surface S. Each edge e of the graph G induces two parallel line-segments (see Figure 1.3a), which may occur either on two disjoint fb-walks or both on the same fb-walk. The operation of ***twisting the edge*** e involves first cutting both line-segments near the same end of e (as in Figure 1.3b), next crossing one segment over the other (as in Figure 1.3c), and finally splicing the ends of the segments (as in Figure 1.3d).

There is a major conceptual difference here from the standard approach to graph imbeddings. In standard topological graph theory (see [21]), twisting an edge means changing the local orientation along the edge and changing the topology of the underlying surface. In our approach here, twisting an edge means introducing a crossing

(a) the two sides of an edge (b) cut ends of the segments

(c) twist the segments (d) reconnect the segment ends

Figure 1.3: Twisting the two sides of an edge

of *strands* on the same surface, but not changing the surface topology.

The correspondence between a graph imbedding $G \to S$ and an immersion of a collection of circuits is illustrated in Figure 1.4. The upper row depicts a planar graph (in (a)) without twisted edges, imbedded on the sphere (in (b)), where each edge is thickened to a band (in (c)), and with the sides of the bands forming a collection of disjoint cycles (in (d)) imbedded on the sphere. The lower row in Figure 1.4 depicts the same planar graph, this time with a set of twisted edges (in (a′)), each marked with a cross) imbedded on the same sphere (in (b′)), with the twisted edges regarded as twisted strip bands (in (c′)), and lastly with the sides of the bands forming a collection of cycles (in (d′)) immersed on the sphere.

Figure 1.5 shows some woven images that have been created using our graphics system [3].

We observe that the bunny and Venus are woven on a simply connected surface, i.e., topologically equivalent to the sphere. However, the teapot and the two multi-toroidal surfaces at the right could not be specified within theories of weaving for the plane, such as that of Grünbaum and Shepard [23].

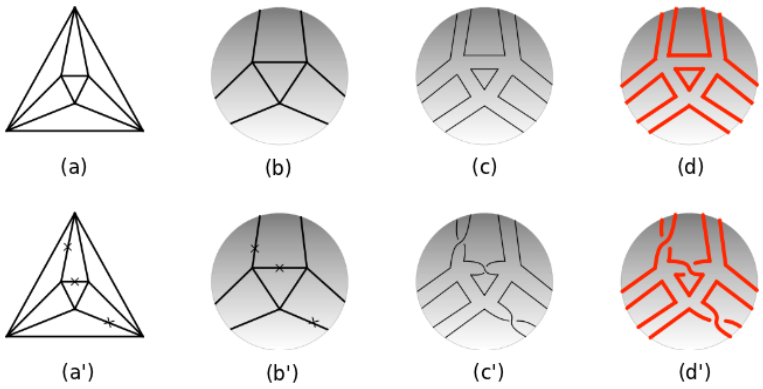


Figure 1.4: The correspondence between an imbedded graph and the induced weaving.

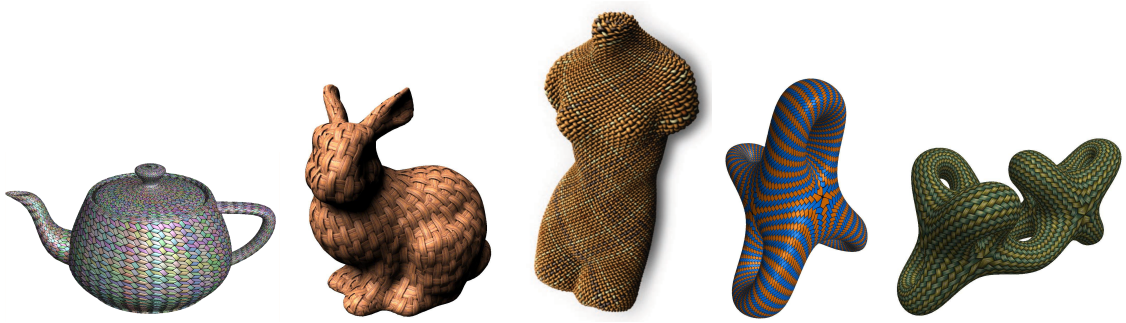


Figure 1.5: Examples of weavings on orientable surfaces

The new development in computer graphics posts a number of interesting questions for theoretical and mathematical studies on graph imbeddings and on weaving structures:

1. What is a formal and appropriate definition of a weaving structure on a topological surface?

2. What graph imbedding with marked twisted edges induces an appropriate weaving structure?

3. What weaving structures are inducible from graph imbeddings with marked twisted edges?

4. Could weavings on low genus surfaces, like Celtic knots on a plane, be induced from an imbedding?

5. What are the relationships between surgery operations on a graph imbedding and that on the induced weaving structure?

6. How to twist the edges of an imbedding graph so that the induced weaving structure achieves specific weaving patterns?

This paper presents a systematic study of these theoretical issues and their applications in computer graphics. We demonstrate how our theoretical system, *extended rotation system* of a graph, models the familiar visual realities of woven objects. We identify the necessary and sufficient conditions for a given weaving on a general surface to be inducible from a graph imbedding on the same surface.

In particular, we prove that any weaving on the plane is inducible from an imbedding of a disconnected planar graph which can be described with a tree structure. Since most Celtic weaves [6] are on the plane, we show that most Celtic weaves could be induced from imbeddings of disconnected planar graphs. Because of the lack of "meaningful" structures to describe disconnected graphs on surfaces with higher genus, our discussion about inducible weaving from disconnected graphs is limited to plane and torus.

Moreover, the practicality of our theoretical system is demonstrated by showing that the two most prevalent subdivision algorithms used in computer graphics, the Catmull-Clark method [10] and the Doo-Sabin method [16], correspond nicely to surgery operations, like *doubling* operation, on *extended rotation systems*. This is

critical to our construction of twill weavings. This study also shows that the doubling operation could refine any weaving and can convert *child-twillable weavings*, some of which are not twillable, into twillable weavings. The necessary and sufficient conditions to identify a child-twillable weaving are also given.

The properties of a child-twillable weaving indicates that a child-twillable weaving consists of only two patterns, and can be represented by a simple *permutation voltage graph*. As an application in computer graphics, a *Quad-Pattern Coverable* mesh[26] is introduced, which consists of only one pattern, and can be represented by a permutation voltage graph with only a bunquet.

At the end of this dissertation, a software example to construct weavings on a plane is described. The software supports two ways of weaving construction: (1) a user could draw a planar graph with an *extended rotation system* to induce a weaving; (2) the software can also reconstruct the weaving with image recognition when a user inputs a high-quality weaving image, which helps to save or modify exisiting weavings in real life.

2. LITERATURE REVIEW

2.1 Weaving Construction

Weaving construction has been studied by mathematicians and artists for hundreds of years[4, 39, 40, 47]. In particular, weaving on a plane is important in both research and practice.

Plane weaving can be constructed from plane graphs. For example, George Bain[6], Iain Bain[7] and Meehan[34, 35, 36, 37] have presented and extended methods to draw Celtic knots based on plane graphs, where the weavings were drew by following the contours of designed plane graphs. Matthew Kaplan and Elaine Cohen[27] also constructed plane weavings from plane graphs: crosses were put on the edges of a plane graph, and then the threads of the crosses were connected to form a weaving.

Plane weavings were also constructed based on planar grids. Peter R. Cromwell[15] designed Celtics knots by adding breaker-markers to control the paths of the strings. He also utilized the underlying symmetries of Celtic knots to extend the design. Jonathna L. Gross and Thomas W. Tucher[22] provided a framework to construct Celtic knots topologically in rectangular grids. In their study, crosses were placed at some lattice-points and connected with each other to form a weaving topologically. Grünbaum and Shepard[23] provided a weaving theory for the plane using combinatorics, number theory and group theory.

Other studies have created weavings with weaving pieces. Mercat[38] composed weaving images by selected weaving patterns, which would be easy to implement but limit the weaving variations. Samuel J. Lomonaco and Louis H. Kauffman[30] constructed all possible weavings by using a set of knot masaics tiles. Cem Yuksel,

Jonathan M. Kaldor, Doug L. James and Steve Marschner[51] provide informal and unlimited rules to generate stitch patterns to compose weavings on 3D meshes.

Weaving construction researches can also been found in knot theory[28, 32].

2.2 Edge Twisting

Paper-strip sculpture[2] provides a method to construct sculptures with paper-trips from imbedded graphs. Each edge in an imbedded graph is viewed as a paper strip, and the two boundaries of each paper-strip are the two half edges of the face boundary walks. The paper-strips are connected with connectors, which correspond to the vertices of the imbedded graph. Furthermore, Akleman et al. [3] noticed that by twisting the paper strips the boundaries of the paper strips define a link in 3D-space, as shown in Figure 2.1. This method provided a fundamental idea to construct weavings from graph imbedding.

(a) (b) (c) (d)

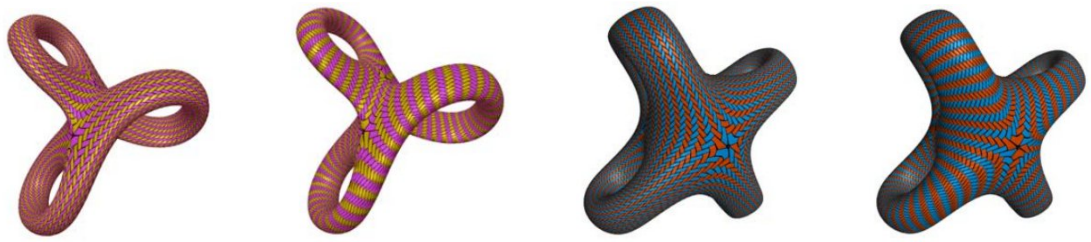
Figure 2.1: The boundary walk of paper-strip sculptures are links[2]

Note that the edge twisting of paper-strips is different from the conventional concepts in the theory of topology[19]. In conventional concepts, edge twisting determines whether a graph is imbedded on an orientable surface or a non-orientable surface. However, in the paper-strip sculpture, the edge twisting operation creates interlaces of circuits without influencing the topological surface.

Akleman et al.[3] provided a definition of *extended rotation system*, which is a *pure rotation system* plus a subset of edges marked as *twisted*. The study showed that an extended rotation system determines a weaving. Interestingly, it proved that if all the edges in the graph are twisted in the same direction (clockwise or counter-clockwise), the obtained weaving would be a plain weaving.

2.3 Twillable Weaving

As an extension, Akelman et al.[1] also constructed twill weaving with the edge-twisting operation, and provided the necessary and sufficient conditions to identify the imbedded graph whose inducing weaving was twillable. However, as not all imbedded graphs can induce a twillable weaving, the study applied two subdivision methods to imbedded graphs: Loop subdivision[31] can always lead to an imbedded graph whose induced weaving was twillable, while Catmull-Clark [10] and Doo-Sabin subdivision[16] subdivisions could not guarantee. Nevertheless, with Catmull-Clark and Doo-Sabin subdivisions, the obtained graph imbeddings could induce a weaving with diagonal patterns(see Figure 2.2).



(a) (b) (c) (d)

Figure 2.2: Diagonal twill pattern on weaving[1]

2.4 Texture Mapping

Another extension of our weaving theory leads to Quad-Pattern Coverable meshes[26], which can be texture mapped with one periodic quad-pattern. As an important area of computer graphics, texture mapping[9] has the advantage of creating complicated images on surfaces based on simple geometry. Among various types of texture mapping, methods with repetitive patterns are especially useful[45], as the repetitive patterns reduce the storing cost and the efforts for texture design.

The main challenge for the texture mapping with repetitive patterns is to maintain the continuity of the textures on the surfaces, which relies on the geometry structure of the mesh and the design of the patterns. For the pattern design, we can classify the repetitive patterns into two types, *periodic pattern* and *aperiodic patterns*. The most popular periodic pattern is the wallpaper pattern on the plane. In 1891, Fedorov put that there are only 17 distinct types of patterns on the plane, which are known as *periodic group* [24] today. Wang tiles[50] is one of the best known aperiodic patterns on the plane. Berger[8] shows the existence of Wang tile by constructing aperiodic sets of 20426 and 104 Wang tiles, and the Wang tile is introduced into computer graphics community by Stam[46] and Cohen et al[13]. In this paper, the repetitive patterns with matching boundaries are sufficient instead of Wang tile, which are similar to image quilting[18].

2.5 Permutation Voltage Graph

As a type of technology to represent symmetric structures, permutation voltage graphs are adopted in this paper. Permutation voltage graphs are developed from Cayley (color) graphs[11] which can generate even-regular graphs based on a finite group and a set of generators. Schreier coset graph [20] generalizes Cayley graphs by

"breaking" the vertex transitive property, and Permutation voltage graph[21] further generalizes Schreier coset graph without limiting the base graph as a bouquet with self-cycles. Thus, the derived graphs in permutation voltage graphs are not necessary to be even-regular. A natural extension of permutation voltage graph is to imbed permutation voltage graphs on surfaces, where the imbedded base graph and the imbeddded derived graph are consistent.

3. PRELIMINARY

Our terminology here is consistent with standard textbooks on topological graph theory (e.g., [21]) and low-dimensional topology (e.g., [33]), but some augmentation is needed.

3.1 2-Manifolds and Surfaces

A ***2-manifold*** is a compact topological space in which each point has a neighborhood homeomorphic to a closed disk. Its ***boundary*** is the set of points that do not have a neighborhood homeomorphic to an open disk; this is necessarily the union of a finite number of mutually disjoint closed curves. A connected 2-manifold is said to be ***closed*** if it has no boundary points; it is ***orientable*** if it does not contain a Möbius band. It is well-known that every orientable closed 2-manifold is homeomorphic to the boundary of a solid multi-toroidal object in Euclidean 3-space; the ***genus of a 2-manifold*** is the number of "handles" in such a solid object. The 2-manifold of genus i is denoted S_i.

A ***surface*** is either a 2-manifold or a connected topological space obtained from a 2-manifold by deleting some of its boundary components. Such a topological space is said to be ***almost compact***. If S is a surface, then

- the surface $brc(S)$ is the result of restoring the missing boundary components;

the surface $brc(S)$ is called the ***boundary-restored compactification*** of S. By way of contrast,

- the surface $int(S)$ is the result of deleting all the boundary components;

it is called the ***interior*** of S. In what follows, a surface is taken to be orientable unless the alternative is declared or inferable from context. A cycle C on a surface S

is ***separating*** if $S \setminus C$ is non-connected, and it is ***contractible*** if it bounds a disk on the surface S. Note that a contractible cycle is also a separating cycle.

3.2 Graph Imbeddings and Graph Rotation Systems

Our ***graphs*** are always undirected. They are connected except when explicit comment or context implies otherwise. Multi-edges and self-loops are allowed. A self-loop has only one endpoint, yet it has two distinguishable ***edge-ends***. To distinguish between the two edge-ends of a self-loop, we regard the interior of each edge as parametrized by the open unit interval (0, 1). The edge-ends are images of small neighborhoods of the limit points 0 and 1, respectively. This distinction permits us to differentiate between the two possible directions in which one can traverse any edge, including self-loops. Two different names can be given to the same endpoint of a self-loop, and interpreted, when context requires, as the two distinguishable edge-ends of the edge. For a multi-edge of multiplicity m, we can use m different names for one of their common endpoints and another m different names for the other endpoint. This allows distinct edges within a multi-edge to have distinguishable names. Under these conventions, each edge in the graph can be given as $e = [v, w]$, with two different ends v and w, and each edge induces two ***oriented edges***, $\langle v, w\rangle$ and $\langle w, v\rangle$, each running from one edge-end of e to the other. A graph G is k***-regular*** if every vertex in G is incident to exactly k edge-ends.

An ***imbedding*** $\iota : G \to S$ is a homeomorphism of the graph G onto a topological subspace of the surface S. The imbedding ι is ***cellular*** if every connected component of $S \setminus \iota(G)$, i.e., the interior of each ***face*** in the imbedding, is homeomorphic to an open disk.[1] Note that cellularity implicitly assumes that the graph G is connected.

[1]Most studies of graph imbedding assume that the imbeddings are cellular. However, in the current paper, we do need to consider non-cellular graph imbeddings. Therefore, an"imbedding" in our discussion can be either cellular or non-cellular.

Definition A ***rotation*** at a vertex v of a graph G is a cyclic ordering of the oriented edges originating at v. Often, when no ambiguity is created, we give the corresponding cyclic ordering of the other endpoints (i.e., other than v) of each of these oriented edges. A ***pure rotation system*** of the graph G consists of a set of rotations, one for each vertex of G. A ***general rotation system*** of G is a pure rotation system of G plus a subset of edges in G that are marked as "twisted".

A ***face corner*** (sometimes, simply ***corner***) in a rotation system ρ_G is a triple (v, e, e'), comprising a vertex v and two oriented edges e and e' oriented out of v, where the oriented edge e' immediately follows the oriented edge e in the rotation at v. The oriented edge e' is said to be 0***-next*** to the oriented edge e at v, and the oriented edge e is said to be 1***-next*** to the oriented edge e' at v.

It is easy to see that a cellular imbedding $\iota_0 : G \to S$ naturally induces a pure rotation system of the graph G. Also, note that every fb-walk in $\iota_0(G)$ is a closed walk, that is, a (cyclically ordered) sequence of oriented edges. Similarly, a cellular imbedding $\iota : G \to S$ of a graph G on a general (orientable or non-orientable) surface induces a general rotation system of the graph G. Conversely, it has been well-known since [25, 17] (for simple graphs) and [21] (for general graphs) that a general rotation system ρ_G of a graph G uniquely determines a cellular imbedding of G on a general (orientable or non-orientable) surface. In particular, if ρ_G is a pure rotation system, then the induced cellular imbedding is on an orientable surface. The surface S can be reconstructed from the rotation system ρ_G by first applying the ***face-tracing algorithm*** that constructs the fb-walks in ρ_G, and then matching the perimeter of a polygon to each fb-walk (see [21]); to an fb-walk of length s, we match an s-sided polygon. In the following discussion, we will interchangeably use the concepts

of "a cellular imbedding of a graph" and "a pure rotation system of a graph". In particular, it is perfectly meaningful to say "a pure rotation system of a graph on an orientable surface".

3.3 Weaving on Topological Surfaces

As in the topological literature [28], a continuous function σ from a collection $\mathcal{C}$ of circuits to a surface S is an ***immersion*** if it is locally one-to-one, i.e., if for any point p in a cycle c in $\mathcal{C}$, there is a neighborhood N_p of p in the cycle c such that the function σ acts homeomorphically from N_p to $\sigma(N_p)$. Note that in the immersion σ, it is possible that the images of two circuits in $\mathcal{C}$ intersect on S, or that the image of a single circuit in $\mathcal{C}$ self-intersects on S. We regard the image of each circuit as a ***strand*** of the weaving. We may refer to the part of a strand between two specified crossings as a ***segment*** of that strand.

Our definition of immersions further requires that every intersection of the images of $\mathcal{C}$ on S be a true intersection, rather than a tangency. The ***thickness of a point*** p on S under the immersion σ is the number of pre-images of p. The ***thickness of the immersion*** σ is the maximum thickness over all points on the surface S.

Our focus in the current paper is on weavings that are ***cyclic***, that is, they are immersions of the union of a set of disjoint circuits on orientable surfaces.

Definition A ***weaving*** on a surface S is an immersion $\sigma : \mathcal{C} \to S$ of thickness at most 2, whose domain $\mathcal{C}$ is a finite collection of circuits[2], in which there are only a finite number of points of thickness 2.

[2]In this paper, both "cycles" and "circuits" refer to simple closed curves. The word "circuits" is used for the closed curves in the domain of a weaving, while the word"cycles" is used for general simple and closed curves, such as those on topological surfaces. The word "strand" is used for the image of a circuit in $\mathcal{C}$.

Definition A ***gap in the weaving*** $\sigma : \mathcal{C} \to S$ is a connected component of $S \setminus \sigma(\mathcal{C})$.

Definition A point of thickness 2 is called a ***crossing***. A vertical order, called a ***crossing-type***, is specified for the pre-images of each crossing to indicate which strand goes over the other at the crossing.

A weaving on a surface naturally induces a graph imbedded on the surface, which is defined as follows.

Definition Let $\sigma : \mathcal{C} \to S$ be a weaving on a surface S. If we regard each crossing in $\sigma(\mathcal{C})$ as a vertex, and each strand segment between two consecutive crossings as an edge, then we obtain a graph G_σ and an imbedding $\iota_\sigma : G_\sigma \to S$. The graph G_σ and the imbedding $\iota_\sigma : G_\sigma \to S$ are called the σ***-graph*** and the σ***-imbedding***, respectively. If a strand has no crossings, then we give it a vertex v and make the strand a self-loop at v.

We remark that a σ-imbedding on a surface S_i need not be cellular. Moreover, the σ-graph need not be connected. For two weavings σ_1 and σ_2 on a surface S_i, if there is an auto-homeomorphism $h : S_i \to S_i$ that maps the imbedded σ_1-graph to the imbedded σ_2-graph, then by appropriately changing the crossing types, we can convert the weaving σ_1 to the weaving σ_2.

3.4 The Extended Edge-Twisting Operation

Edge-twisting operations have been used extensively by topological graph theorists in the study of graph imbeddings on non-orientable surfaces. We now describe

how the concept has been nicely borrowed and extended in computer graphics for the construction of weavings [3].

In a topological understanding of graph theory, traversing a twisted edge "reverses" the local orientation of the rotation system. Accordingly, a re-twisting of a twisted edge is equivalent to untwisting the edge, so the result of double-twisting an edge is topologically equivalent to no twisting at all. By way of contrast, in our model for weaving, the two trace-pairs induced by an untwisted edge are regarded as two parallel segments of woven strands— and twisting the edge is interpreted as crossing the two strands. We are also interested in knowing which strand goes over and which strand under at a crossing point, and by how many turns a strand segment is twisted around the other segment. Double-twisting an edge is not the same as leaving it untwisted. Figure 3.1 gives some intuitive illustrations for edge-twisting in terms of the above interpretation.

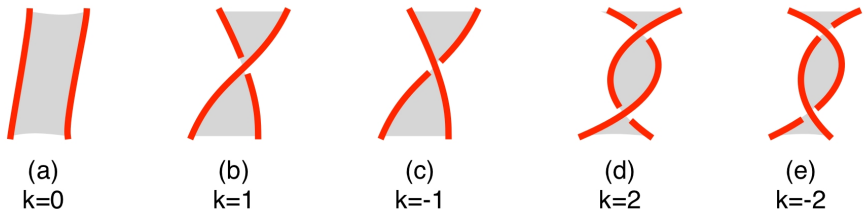


Figure 3.1: (a) an untwisted edge. (b) a counterclockwise twisted edge. (c) a clockwise twisted edge. (d) a double-counterclockwise twisted edge. (e) a double-clockwise twisted edge.

Comparing (b) with (c) and (d) with (e) in Figure 3.1 reveals that the direction in which we twist the edge is clearly relevant to which segment is over the other. This motivates the following definitions.

Definition An edge is 1^+***-twisted*** (resp. 1^-***-twisted***) if it is obtained from a flat paper strip, whose two sides are interpreted as the two parallel segments induced by the edge, by imagining yourself as positioned at one end of the strip, and twisting that end in counterclockwise (resp. clockwise) direction by 180°, while fixing the opposite end, assuming that you are facing the clock. There is an alternative characterization of a 1^+-twisted edge may make it easier for some readers to visualize. As you walk along the corresponding flat strip, the edge to your left crosses over the edge to your right. We observe that this characterization is independent of the direction in which you traverse the strip.

See Figures 3.1(b,c) for an illustration. This concept can be naturally extended to multiple twists: we say that an edge is k^+***-twisted*** (resp.k^-***-twisted***) for an integer $k \geq 0$ if the edge can be obtained from an untwisted edge by k consecutive 1^+-twists (resp. 1^--twists). See Figures 3.1(d,e). In topological graph theory [21], a k^+-twisted or k^--twisted edge is equivalent to an untwisted edge if k is even, and equivalent to a normally twisted edge if k is odd.

Definition An ***extended rotation system*** (abbr. ***ERS***) for a graph G is obtained from a pure rotation system of G by assigning a number k of twists, with $k \in \mathbb{Z}$, to every edge of G. (Elsewhere we have called this an "extended general rotation system".)

Let ρ_G^0 be a pure rotation system of a graph G, which induces an imbedding $G \to S$. Assigning an edge-twist integer to every edge in G results in an ERS ρ_G of G. As explained above, the ERS ρ_G induces a weaving $\sigma : \mathcal{C} \to S$ on the surface S.

Moreover, if we take evenly twisted edges as untwisted and oddly twisted edges as normally twisted, and if we let $\hat{\rho}_G$ denote the resulting (non-extended) general rotation system, then the classical ***face-tracing algorithm*** applied to the rotation system $\hat{\rho}_G$ produces a set of fb-walks [21].

It is easy to verify that these fb-walks of $\hat{\rho}_G$ are ***exactly*** the set of strands of the induced weaving σ. Also, based on the edge-twisting assignments in the ERS ρ_G, we can determine precisely which strand overcrosses and which undercrosses at each crossing, and how many times it crosses. Therefore, the face-tracing algorithm can be revised, with only small changes, to become a ***strand-tracing algorithm*** that produces all the strands of the induced weaving σ and provides complete information for the strand crossings.

For the convenience of our discussion, we present strand-tracing as Algorithm 1, where for an oriented edge $\langle u, w\rangle$ and $t \in \{0, 1\}$, $type([u, w])$ is the twist-value assigned to the corresponding edge $[u, w]$, and the function $\text{Next}(\langle u, w\rangle, t)$ yields the oriented edge that is t-next to the oriented edge $\langle u, w\rangle$ at u. We say that a face corner (u, e, e') is "untraced" if no trace has followed the inverse of the oriented edge e while entering the vertex u, and then left u along the oriented edge e'.

Algorithm StrandTrace(ρ_G)
Input: an ERS ρ_G for a graph G.
Output: the set of strands for the weaving induced by ρ_G.

while there is an untraced face corner (u, e, e') in ρ_G **do**
call Strand($\langle u, w\rangle, 0$). \\ assuming $e' = \langle u, w\rangle$

Subroutine Strand($\langle u_0, w_0\rangle, t_0$)
\\$\langle u_0, w_0\rangle$ is an oriented edge, $t_0 \in \{0, 1\}$ is the "trace type".
1. trace $\langle u_0, w_0\rangle$; $t = t_0 + type([u_0, w_0])$ $(mod\ 2)$;
2. $\langle u, w\rangle = \text{Next}(\langle w_0, u_0\rangle, t)$; \\$u = w_0$
3. **while** $(\langle u, w\rangle \neq \langle u_0, w_0\rangle)$ or $(t \neq t_0)$ **do**
{ trace $\langle u, w\rangle$; $t = t + type([u, w])$ $(mod\ 2)$; $\langle u, w\rangle = \text{Next}(\langle w, u\rangle, t)$ }.

Algorithm 1: The strand-tracing algorithm.

4. CELLULAR WEAVING AND NORMAL WEAVING

Cellular imbedded graphs have been extended studied in topological graph theory[21], which are also meaningful and are widely used in practice. For example, a mesh, which is used for shape understanding and object representation in computer graphics[43], is a cellular imbedding on an orientable surface. In this chapter, we study the properties of an induced weaving from an ERS on a cellular imbedded graph.

4.1 Cellular Weavings

We derive a characterization of *cellular weavings* in terms of extended rotation systems on graphs.

Definition A weaving $\sigma : \mathcal{C} \to S$ is ***cellular*** if every gap in σ is homeomorphic to an open disk.

A woven object would look quite strange if unwoven patches of structural surface bulged out of the weave or if the weave could be pulled apart without tearing it. Theorem 4.1.1 (which is routine for topological graph theorists) explains that cellular weavings avoid such undesirable properties.

Theorem 4.1.1 *Let $\sigma : \mathcal{C} \to S_i$ be a non-cellular weaving. Then either there is a separating cycle in the surface S that separates σ into two disjoint non-empty weavings, or the weaving σ can be implemented as an immersion $\sigma : \mathcal{C} \to S_{i-j}$, where $j \geq 1$.*

PROOF. Since σ is not cellular, there is a gap g that is not homeomorphic to an

open disk. Then the gap g contains a closed curve c that is not contractible in g. There are two cases.

First we suppose that the closed curve c separates the surface S_i into two surfaces-with-boundary, $S_j - int(D)$ and $S_{i-j} - int(D')$, where D and D' are closed disks. If the woven image $\sigma(\mathcal{C})$ intersects both $S_j - int(D)$ and $S_{i-j} - int(D')$, then the weaving σ is separable into two disjoint non-empty weavings, one on S_j and the other on S_{i-j}. Otherwise, if $\sigma(\mathcal{C}) \cap S_j = \emptyset$ (a similar analysis applies when $\sigma(\mathcal{C}) \cap S_{i-j} = \emptyset$), then $S_j - int(D)$ cannot be homeomorphic to an open disk – lest the closed curve c be contractible in the gap g, which is contrary to supposition. Therefore, the image of the weaving σ is entirely contained in S_{i-j}, and S_j is an orientable surface of positive genus. Thus, the weaving σ is interpretable as an immersion from the circuit collection $\mathcal{C}$ into the surface S_{i-j}, with $j \geq 1$.

Alternatively, if the closed curve c is non-separating on the surface S_i, then cut the surface S_i along c, and fill the two resulting holes with open disks, to obtain the surface S_{i-1}. The weaving σ is now interpretable as an immersion of the circuit collection $\mathcal{C}$ on the surface S_{i-1}. □

To understand which extended rotation systems of a graph G induce cellular weavings, we start with Lemma 4.1.2 and Corollary 4.1.3, which can be easily verified. It asserts that a weaving induced by an arbitrary ERS of a graph can also be induced by a ***tri-valued rotation system*** (abbr. ***TRS***) in which only 0-twisted,1^+-twisted, and 1^--twisted edges are allowed, if the graph G is replaced by a subdivision of itself.

Lemma 4.1.2 *Let $\sigma : \mathcal{C} \to S_i$ be a weaving induced by an ERS ρ_G of a graph G, and let e be an edge in ρ_G that is k^+-twisted (resp. k^--twisted) with $k > 1$. Then the weaving σ is also induced by the ERS that is obtained from ρ_G by replacing the edge e with a path of k 1^+-twisted (resp. 1^--twisted) edges.*

Corollary 4.1.3 *Let $\sigma : \mathcal{C} \to S_i$ be a weaving induced by an ERS ρ_G of a graph G. Then the weaving σ is also induced by a tri-valued rotation system.*

4.1.1 Medial graphs

Two gaps in a weaving $\sigma : \mathcal{C} \to S_i$ are ***adjacent*** if they share a common strand segment. The weaving σ is ***2-colorable*** if its gaps can be colored with two colors, such that no two adjacent gaps are assigned the same color.

Figure 4.1(a) illustrates an imbedding $K_4 \to S_0$ induced by a pure rotation system on K_4. Figure 4.1(b) is a redrawing of 5(a) so that the strands of the weaving are pulled out of the corners. In the ERS of that figure, every edge of the graph K_4 is twisted. The weaving induced by the Strand-Tracing Algorithm has three strands, which are colored blue, green, and red. There are eight gaps, four of which, called ***vertex-gaps***, are neighborhoods of the four vertices, and four of which, called ***face-gaps***, lie in the interiors of the four faces of the pure rotation system. We observe the following two properties.

- Each of the gaps is a 2-cell.
- The adjacencies of each vertex-gap are only to face-gaps, and each face-gap is adjacent only to vertex-gaps.

Theorem 4.1.6 establishes that these two properties always hold when all the edges are twisted. Introducing the concept of *medial graph* paves the way to a short proof.

Let $G \to S_i$ be a cellular imbedding. The ***medial graph*** is constructed (e.g., see [5]) according to Algorithm 2. We recognize the white vertices and the colored edges in Figure 4.1(b) as the medial graph for the given imbedding $G \to S_0$.

Proposition 4.1.4 (see Theorem 2.1 of [5]) *Let $G \to S_i$ be an imbedding of a 4-regular graph whose faces can be properly 2-colored. Then $G \to S_i$ is the medial*

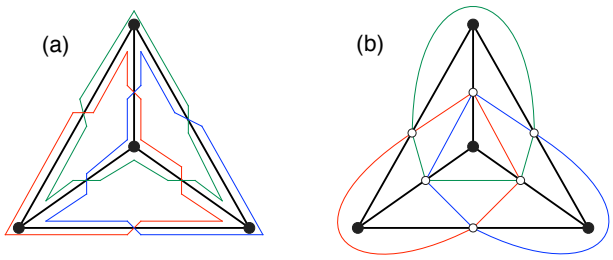


Figure 4.1: (a) The graph K_4 with an ERS and the induced weaving. (b) A multi-color medial graph for the imbedding $K_4 \to S_0$.

Algorithm MedialGraph(ρ_G^0)
Input: a pure rotation system ρ_G^0 for a graph G.
Output: an imbedding of the corresponding medial graph.

(S0) Use an applied face-tracing (see [21]) on the pure rotation system ρ_G^0 to construct an imbedding $G \to S_i$.
(S1) For each edge $e \in E_G$,
Install a medial vertex w_e in the interior of e —
this vertex is usually drawn at or near the middle of e.
(S2) For each corner (v, e, e') of each face f of the imbedding $G \to S_i$,
Insert an edge $[w_e, w_{e'}]$ within the face f.
(S3) For each edge $e = [u, v] \in E_G$, we suppose that the corners (u, e', e) and (v, e, e'') lie on one side of e and that the corners (v, d', e) and (u, e, d'') lie on the other side.
The induced rotation at the medial vertex w_e is w_e. $w_{e'}$, $w_{e''}$, $w_{d'}$, $w_{d''}$

Algorithm 2: The medial-graph algorithm.

graph imbedding for a unique dual pair of graph imbeddings in S_i.

An ERS for a graph is ***simply-twisted*** if every edge is either 1^+-twisted or 1^--twisted.

Proposition 4.1.5 *Let ρ_G be a simply-twisted rotation system, such that the corresponding pure rotation system induces the graph imbedding $G \to S_i$, and let σ be the*

induced weaving. Then the σ-imbedding $\iota_\sigma : G_\sigma \to S_i$ is equivalent to the imbedding of the medial graph for $G \to S_i$.

Theorem 4.1.6 *A 2-colorable weaving $\sigma : \mathcal{C} \to S_i$ is cellular if and only if it is inducible from a simply-twisted rotation system of a graph.*

PROOF. Let $\iota_\sigma : G_\sigma \to S_i$ be the induced 2-colorable, cellular σ-imbedding. The graph G_σ is clearly 4-regular, and a facial 2-coloring is inherited from the weaving $\sigma : \mathcal{C} \to S_i$. By Proposition 4.1.4, there is a graph imbedding $G \to S_i$ for which $\iota_\sigma : G_\sigma \to S_i$ is the medial imbedding. Take ρ_G to be the ERS whose pure rotation system corresponds to the imbedding $G \to S_i$, with every edge either 1^+-twisted or 1^--twisted, so as the reproduce the overcrossings and undercrossings of $\sigma : \mathcal{C} \to S_i$.

Conversely, let ρ_G be a simply-twisted rotation system of a graph G. The medial graph of the imbedding $G \to S_i$ induced by the corresponding pure rotation system is cellular. We observe that some faces of the medial graph imbedding lie within a face of the imbedding $G \to S_i$, and that each of the others contains a vertex of G. If the former are colored with one color and the latter with another color, then the result is a 2-coloring of the map of the medial imbedding. □

4.1.2 Topological edge-contraction

Let ρ_0 be a pure rotation system of a graph G on a surface S_i and let e be an edge of G. ***Contracting the edge*** e ***(topologically) in (the imbedding induced by) the rotation system*** ρ_0 means to continuously shrink the edge e on the surface S until its two ends meet. Any self-loops or multiple edges that result from this are retained in topological contractions, in which respect they differ from the "combinatorial contractions" that occur in the theory of simple graphs. Let $e = [u, v]$, and let the endpoints u and v in $\rho_0(G)$ have the following rotations,

respectively:

$$u : u_1, \ldots, u_{i-1}, v, u_{i+1}, \ldots, u_s; \qquad v : v_1, \ldots, v_{j-1}, u, v_{j+1}, \ldots, v_t. \tag{4.1}$$

Contracting the edge in ρ_0 replaces the two vertices u and v with a new vertex w whose rotation is

$$w : u_1, \ldots, u_{i-1}, v_{j+1}, \ldots, v_t, v_1, \ldots, v_{j-1}, u_{i+1}, \ldots, u_s. \tag{4.2}$$

It is well-known in topological graph theory that contracting an untwisted edge that is not a self-loop in an (orientable or non-orientable) imbedding does not change the imbedding surface. We can generalize edge contraction to extended rotation systems of a graph.

Definition Let ρ be an ERS of a graph G consisting of a pure rotation system ρ_0 and an edge-twist assignment. Let e be an edge in G. Then ***contracting the edge*** e ***in the ERS*** ρ results in a new ERS, whose pure rotation system is obtained by contracting the edge e in ρ_0, with edge-twist assignments on all other edges identical to that in $\rho(G)$.

Theorem 4.1.7 *Let ρ be an ERS of a graph G, and let ρ' be the ERS obtained by contracting a 0-twisted edge e in ρ, where e is not a self-loop. Then the ERS ρ and the ERS ρ' induce equivalent weavings, on the same surface.*

Proof. Let σ be the weaving induced by the ERS ρ. Since the edge e is 0-twisted, it is easy to verify that the edge e in $\rho(G)$ is entirely contained in a single vertex-gap in the weaving σ. Contracting the edge e in $\rho(G)$ contracts a line segment

(corresponding to the edge e) in that vertex-gap. This changes neither the surface topology nor the gap structure in σ. Therefore, the ERS $\rho'(G')$ induces equivalent weavings, on the same surface. □

Let E_1 be a subset of edges of a graph G. Let $G(E_1)$ denote the subgraph of G whose vertex set consists of the end-vertices of the edges in E_1 and whose edge set is E_1.

Corollary 4.1.8 *Let ρ be an ERS, and let ρ' be an ERS obtained by contracting a set E_1 of 0-twisted edges in $\rho(G)$, such that the subgraph $G(E_1)$ is acyclic. Then the ERS ρ and the ERS ρ' induce equivalent weavings, on the same surface.*

Combining Corollary 4.1.8 and Theorem 4.1.6, we obtain

Theorem 4.1.9 *A 2-colorable weaving is cellular if and only if it is inducible by an ERS ρ of a graph G in which the subgraph $G(E_1)$ contains no cycles, where E_1 is the set of all 0-twisted edges in ρ.*

4.2 Normal Weaving

Cellular weaving requires that every gap is homeomorphic to an open disk, which is sometimes too restrictive. Indeed, it is not rare that an artistic weaving pattern consists of several disjoint and unlinked pieces (in which case the σ-graph is not connected), so that the weaving cannot be cellular. Figure 4.2 gives some examples of such weaving patterns. In this section, we characterize a more general class of weavings, called *normal weavings*, as inducible from extended graph rotation systems. For instance, this characterization implies that all the weavings in Figure 4.2 are inducible from extended graph rotation systems.

Figure 4.2: Examples of non-cellular weaving patterns (from [49])

We recall that a weaving is 2-***colorable*** if its gaps can be colored with 2 colors such that no two adjacent gaps have the same color. Such a coloring is called a 2-***coloring of the weaving***.

Definition A weaving is ***normal*** if it has a 2-coloring in which every gap in one of the two colors is homeomorphic to an open disk. It is clear that a cellular weaving is a special case of a normal weaving.

Theorem 4.2.1 *Let ρ_G be an ERS of a graph G. Then the induced weaving σ is normal.*

Proof. The proof is somewhat similar to one direction of the proof of Theorem 4.1.6. The graph G continues to be taken to be connected, but we now allow some 0-twisting in the ERS ρ_G, as well as 1^+- or 1^--twisting. Invoking Lemma 4.1.2 once again enables us to ignore higher multiplicity twisting.

We suppose that the pure rotation system ρ_G^0 imbeds the graph G on an orientable surface S_i, so that the weaving σ induced by ρ_G is a weaving on the surface S_i. We construct the weaving $\sigma : \mathcal{C} \to S_i$ with the algorithm **NormalWeave** (Algorithm 3). Figure 4.3 illustrates why the weaving σ is normal.

NormalWeave (ρ_G)
Input: an ERS ρ_G for a graph G.
Output: a normal weaving on the surface induced by the ERS ρ_G.

(S0) Use face-tracing (see [21]) to construct the orientable imbedding $G \to S$
based on the pure rotation system ρ_G^0.
(S1) For each oriented edge $\hat{e}$ induced by an edge e on an fb-walk f in ρ_G^0:
(S1.1) if e is 1^+- or 1^--twisted, then place a crossing $x_{\hat{e}}$ at the middle of e —
this crossing should preserve the crossing-sense of ρ_G;
(S1.2) if e is 0-twisted, then place a point $x_{\hat{e}}$ in the face f near the middle of e.
\\ Thus, if e is 1^+- or 1^--twisted, then for the two oriented edges $\hat{e}$ and $\hat{e}'$ induced by e,
\\ then we have $x_{\hat{e}} = x_{\hat{e}'}$.
(S2) For every pair of oriented edges $\hat{e}_1$ and $\hat{e}_2$ that are consecutive along the fb-walk f in ρ_G^0,
insert an edge $[x_{\hat{e}_1}, x_{\hat{e}_2}]$ within the face f.
(S3) Remove all vertices and edges that are in the original graph G; and
(S4) If a vertex $x_{\hat{e}}$ has degree 2, then smooth it.

Algorithm 3: The normal weaving algorithm.

To see why such a weaving has a 2-coloring, we consider how it evolves from the 2-colorable weaving σ^0 (illustrated with thin red lines in Figure 4.3(a)) associated with its pure rotation system ρ_G^0. In the weaving σ^0, there is one vertex gap, which amounts to a thickening of the entire graph. Since ρ_G^0 induces a cellular graph imbedding, all the face-gaps are 2-cells. Twisting an edge may cause some minor reshaping of face-gaps, but it does not merge them, affect their cellularity, or cause them to become adjacent to other face-gaps. Twisting an edge may split a vertex-gap into two vertex-gaps or change the connectivity of a vertex-gap, but it does not cause two vertex-gaps to become adjacent. Accordingly, the output of Algorithm 3 is a normal weaving. □

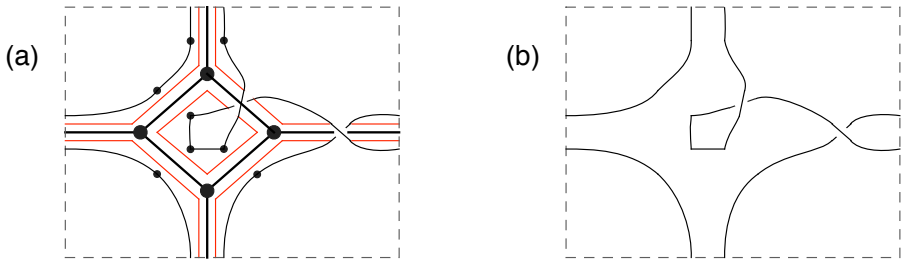


Figure 4.3: (a) Incipient induced weaving for an ERS, after Steps (S0), (S1), and (S2). (b) Induced weaving, after Steps (S3) and (S4).

Proving the converse of Theorem 4.2.1 is more complicated.

The graph with one vertex and ℓ self-loops is called the ***bouquet*** of ℓ self-loops and denoted B_ℓ.

Theorem 4.2.2 *Let $\sigma : \mathcal{C} \to S_i$ be a normal weaving on a surface S_i. Then there is a graph G and an ERS ρ_G that induces σ.*

PROOF. We bipartition the gaps of the weaving $\sigma : \mathcal{C} \to S_i$ into F-gaps and V-gaps, such that no two gaps in the same group are adjacent, and so that every F-gap is homeomorphic to an open disk. (F-gaps and V-gaps will turn out to be face-gaps and vertex-gaps, respectively.)

The boundary-restored compactification of each V-gap m is a compact surface of some genus g with some number $k \geq 1$ of boundary components, each of which corresponds to one of the fb-walks of the imbedding $\iota_\sigma : G_\sigma \to S_i$ of the graph of the weaving $\sigma : \mathcal{C} \to S_i$. (All of the boundary components of a V-gap are missing.) The boundary-restored compactification $brc(m)$ of the surface m is homeomorphic (see [33]) to the result of pairing the sides of a $4g$-sided flat polygon with k disjoint holes in its interior, so that the image of the polygon boundary is homeomorphic to B_{2g}. Accordingly, we can imbed the bouquet B_{2g} in the interior of the surface m so that

every region is planar.

When $g = 0$, this means that we have placed the trivial graph B_0 in the interior of m. We conceptualize the "degenerate" 0-sided polygon with one vertex as the "flat polygon" for $brc(m)$.

We now add $k - 1$ more self-loops to this bouquet, each drawn so that it begins and ends at a corner of the flat polygon, so that there is a single hole in each region. The edges in the graph B_{2g+k-1} are called "dividing edges." See Figure 4.4(a) for two representations of a gap with genus $g = 2$ and $k = 2$ holes.

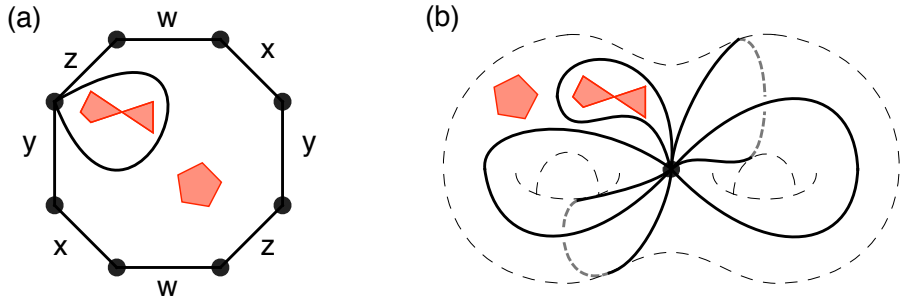


Figure 4.4: Two representations of a gap of genus 2 with two holes. (a) An 8-sided polygon with two (red) holes. (b) The corresponding imbedding of the bouquet B_5 in a two-holed surface of genus 2.

Our immediate task is construction of the graph G to be used for inducing the given normal weaving σ. We start with a single face f of the imbedding $\iota : B_{2g+k-1} \to m$, which contains exactly one hole. The boundary of that hole is a simple cycle in the compact surface $brc(m)$. We emphasize here that all of the boundary components of a gap are missing. Thus, when two F-gaps that are neighbors of a given V-gap meet each other at a vertex, the result of deleting the union of the boundaries of the two F-gaps is a single hole in the gap m, which we may regard as polygonal, with

each side of the polygon corresponding to an edge of the graph G_σ. Figure 4.5(a) illustrates the face inside the monogon of the imbedding of B_5 in Figure 4.4(a). We observe that the vertex at which two F-faces meet has been virtually "pulled apart", which is how it seems from the perspective of the V-gap.

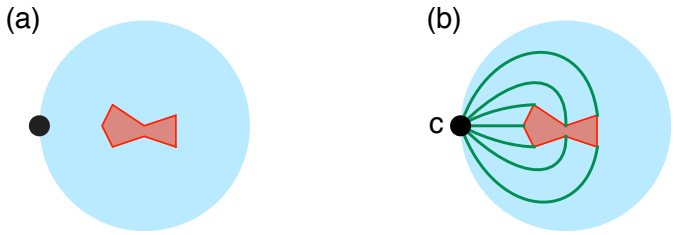


Figure 4.5: (a) Virtual merging of a 3-sided hole and a 4-sided hole into a 7-sided hole. (b) Seven "semi-edges" from the corner c to the seven corners of the polygonal hole, and the seven resulting "semi-faces".

Now choose any corner c of face f, and draw a set of internally disjoint simple curves on face f from the corner c to the corners of the polygonal hole inside face f, one to each polygon corner, as in Figure 4.5(b). These curves are called "semi-edges", and the resulting faces within face f are called "semi-faces". (It will soon be clear why we name these objects so.) We construct semi-edges similarly in every face of the gap m and we call the result G_m. We repeat this construction in every face of every V-gap. We consider the union

$$\bigcup_{\text{all V-gaps } m} G_m$$

over all V-gaps in the weaving $\sigma : \mathcal{C} \to S_i$. Since each crossing in σ is on exactly two V-gap corners (which may belong to the same gap), there are exactly two semi-edges meeting at that crossing. We weld two such semi-edges into a single edge, which we

call a "connecting edge". We denote the resulting graph by G and its imbedding in S_i by $\iota : G \to S_i$. We will show that an ERS of the graph G based on the imbedding $\iota : G \to S_i$ induces the weaving σ.

Each face x of the imbedding $\iota : G \to S_i$ is the union of a single s-sided polygonal F-gap of the weaving $\sigma : \mathcal{C} \to S_i$ with the s semi-faces that meet its s sides, obtainable by pasting each of the semi-faces to the F-gap across one of the sides of the F-gap. Since the F-gap is cellular, and since each of the semi-faces is cellular, it follows that the face x is cellular. Accordingly, the imbedding $\iota : G \to S_i$ is cellular. This implies that the graph G is connected.

Let ρ_G^0 be the pure rotation system of the connected graph G corresponding to the cellular imbedding $\iota : G \to S_i$. Let ρ_G be the ERS of the graph G that consists of the pure rotation system ρ_G^0 and the edge-twist assignment such that all dividing edges in G are 0-twisted and the connecting edges in G are 1^+-twisted or 1^--twisted, so as to be consistent with the weaving σ, as illustrated in Figure 4.6(a,b).

Figure 4.6: Twisting a connecting edge (green) to be consistent with the given weaving (red).

Let σ_G be the weaving on the surface S_i that is induced by the ERS ρ_G. By construction, the weaving σ_G has a crossing wherever the given weaving σ has a crossing, and only where σ has a crossing. This implies a bijection between the set of strands of σ and the set of strands of σ_G. Moreover, each crossing of σ_G is of the

same type as the corresponding crossing of σ. It follows that the two weavings are equivalent. □

Theorem 4.2.1 and Theorem 4.2.2 combine into the following:

Corollary 4.2.3 *A weaving σ is inducible by an ERS of a graph if and only if σ is a normal weaving.*

We develop cellular weavings and normal weavings from ERSs on cellular imbedded graphs above, and from Corollary4.2.3 we also show that any weaving induced from cellular imbeddings is normal, which is a limitation for an ERS on a cellular imbedding. We would like to extend the limitation under the same framework.

5. WEAVINGS ON SPHERE AND TORUS

As indicated in the last chapter, a weaving induced from a connected graph is normal. In this chapter, we try to break the normal weaving limitation by extending our framework to imbeddings of disconnected graphs, which are non-cellular imbeddings. Since a non-cellular imbedding is not simply determined by its rotation system[19], it is difficult to describe a non-cellular imbedding on surfaces other than sphere and torus mathematically. We mainly discuss imbeddings of disconnected graphs in low genus surfaces, like sphere and torus, with the assistances of auxiliary information.

5.1 Plane Weaving

We first consider the weavings on the plane. A weaving $\sigma : \mathcal{C} \to S$ is a ***plane weaving***, if the circuits $\mathcal{C}$ are immersed on a plane (or equivalently, on a sphere). A graph G is a ***planar graph***, if G can be drawn in the plane where the images of two different edges only meet at vertices. If not specified, a planar graph may be connected or disconnected. A planar graph imbedded in a plane is called ***plane graph***. A weaving σ is ***connected***, if its σ-graph is connected; Otherwise, the weaving σ is ***disconnected***.

Let G be a planar graph, and $\iota(G)$ be a plane graph. There is only one unbounded region in plane graph $\iota(G)$ and is called ***exterior*** region. Two rotation systems $\iota_1 = \{\iota_1(v) \mid v \in V(G)\}$ and $\iota_2 = \{\iota_2(v) \mid v \in V(G)\}$ of the same graph G are ***equivalent***, if their rotations $\iota_1(v)$ and $\iota_2(v)$ are the same for each vertex $v \in V(G)$. Note that two plane graphs with equivalent rotation systems may not be topologically equivalent, since their exterior region may choose differently.

Let G be a connected planar graph, and c_1, c_2 be the cycles bounding the ex-

terior region of plane graphs $\iota_1(G)$ and $\iota_2(G)$. Then $\iota_1(G)$ and $\iota_2(G)$ are ***plane-isomorphic***, if there is an isomorphism between $\iota_1(G)$ and $\iota_2(G)$ which maps c_1 to c_2.

It is easy to understand the meaning of plane-isomorphism. Let $\iota'_1(G)$ and $\iota'_2(G)$ be two imbeddings of a graph G with the same rotation systems on a sphere. It is well known that a planar graph can be cellularly imbedded on a sphere and the imbedding is uniquely determined by a rotation system. Thus, the two imbeddings $\iota'_1(G)$ and $\iota'_2(G)$ on a sphere are isomorphic. Let c_1, c_2 be face-boundary walks of two faces in $\iota'_1(G)$ and $\iota'_2(G)$. The faces f_1, f_2 bounded by c_1 and c_2 are the same face, up to isomorphism. Therefore, by removing points p_1, p_2 from the inside of f_1 and f_2, the two plane graph $\iota_1(G)$ and $\iota_2(G)$ are plane-isomorphism.

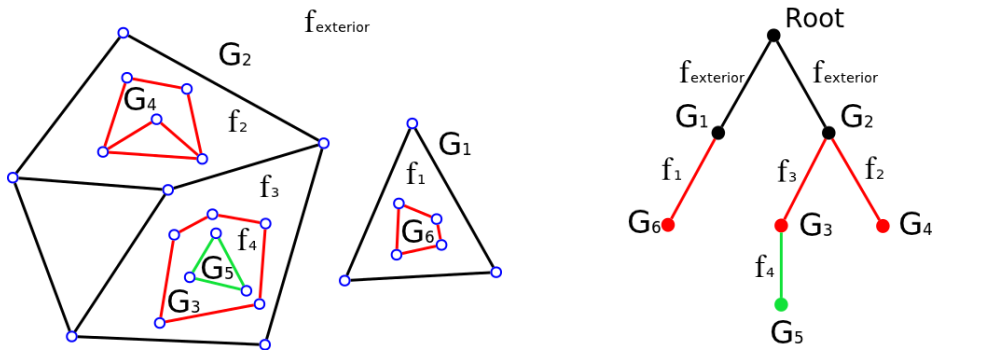
5.1.1 Weaving from plane graph

According to Theorem 4.2.3, an ERS of a connected graph induces a normal weaving on a surface. In the following, we focus on the weaving induced from a disconnected plane graph.

Let $\iota(G)$ be the plane graph with a pure rotation system $\rho^0(G)$. Assume $\iota(G)$ contains connected components $\iota(G_1), \iota(G_2), \ldots, \iota(G_k)$ with $k > 1$. We say a component $\iota(G_i)$ is a ***child component*** of another component $\iota(G_j)$, if $\iota(G_i)$ is entirely contained in an inner region f of $\iota(G_j)$ and there is no other component $\iota(G_t)$ such that $\iota(G_t)$ is also contained in f and $\iota(G_i)$ is contained in an inner region of $\iota(G_t)$.

To describe the structure of a disconnected plane graph $\iota(G)$, we introduce a ***tree of a plane graph*** T_G with a virtual empty root, where each node contains a component of $\iota(G)$. We set the components, which are not children of any other components, as children of the root, and we insert the other components of $\iota(G)$ with the parent-child relationships to create T_G. We label the edges of the tree with

information to indicate in which inner region a given child is contained. Figure 5.1 gives an example of tree structure.



(a) A disconnected plane graph (b) The tree of the plane graph

Figure 5.1: An example of the tree of a plane graph

In the rest of this chapter, we extend the definition of ***imbedding a disconnected planar graph*** G by including the tree structures T_G. Thus, for a disconnected planar graph G, two imbeddings $\iota_1(G)$, $\iota_2(G)$ are different if they have the same rotation systems but non-isomorphic tree structures. Similarly, we also extend the definition of extended rotation system for disconnected planar graphs G by including the tree structure T_G.

Lemma 5.1.1 *Given an ERS ρ of disconnected planar graph G, the induced weaving σ is a plane weaving.*

Proof. Assume the connected components of $\iota(G)$ are $\iota(G_1), \iota(G_2), \ldots, \iota(G_k)$, with $k > 1$. By twisting the edges along the twist-types, we obtain the induced plane weaving $\sigma_1, \sigma_2, \ldots, \sigma_k$ respectively. Let $\iota(G_i)$ be a child of $\iota(G_j)$, where $\iota(G_i)$ is entirely contained in an inner region of $\iota(G_j)$. After twisting the edges, the weaving

σ_i is still entirely contained in an inner gap of σ_j. The weaving σ composed of disjoint plane weavings is also a plane weaving. □

5.1.2 Plane graph from plane weaving

In this section, we construct plane graphs from given plane weavings. Since the σ-imbedding of a plane weaving is a 4-regular plane graph, the gaps of a plane weaving σ is 2-colorable. In the following, we will frequently use the 2-colorable property of a plane weaving.

We first consider a connected weavings σ on the plane. Recall that an ERS of a graph is simply-twisted if every edge of the graph is either 1^+-twisted or 1^--twisted.

Theorem 5.1.2 *Let $\sigma : \mathcal{C} \to \mathbb{R}^2$ be a connected plane weaving. Then σ is a cellular weaving, and it is inducible by a simply-twisted ERS of a planar graph.*

Proof. Each vertex of the σ-graph G_σ represents a crossing of the weaving σ and has degree 4. Therefore, the σ-imbedding $\iota_\sigma : G_\sigma \to \mathbb{R}^2$ induces a plane map in which every vertex is 4-valent. By Kempe's two-color map theorem [44], such a map is 2-colorable. Accordingly, the weaving σ is 2-colorable. Moreover, the σ-imbedding $\iota(G_\sigma)$ is cellular, because a planar imbedding of any connected graph is cellular. Thus, by Theorem 4.1.6, the weaving σ is inducible by a simply-twisted ERS of a planar graph. □

Then, another question arises:

> *Given a disconnected plane weaving σ, can we induce σ by an ERS of a planar graph?*

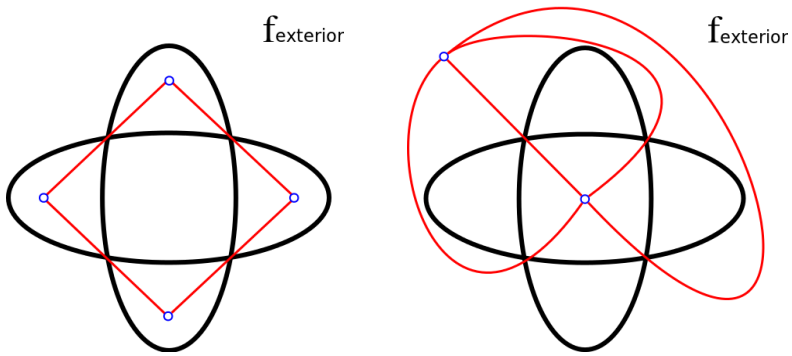
We first define two types of corresponding graphs for a connected plane weaving σ.

1. Let the planar graph G which can induce σ with a simply-twisted ERS $\rho(G)$ be the ***F-planar graph*** of σ, where the imbedded graph $\iota(G)$ under the pure rotation system $\rho^0(G)$ is the ***F-plane graph*** of σ. The algorithm to construct F-plane graph is given in Figure 5.3.

2. Let the dual graph of the F-plane graph be the ***V-plane graph*** of σ, where its underlying graph is the ***V-planar graph***.

Figure 5.2 gives an example of the F-plane graph and V-plane graph of the same weaving.

Note that the medial graph of the V-plane graph $\iota'(G_d)$ is not plane-isomorphic to the σ-imbedding ι_σ, since the medial graph keeps each vertex of $\iota(G_d)$ in a inner gap, while the exterior vertex should correspond to the exterior gap for the plane-isomorphism. However, the medial graph of the imbedded graph $\iota_s(G)$ and $\iota_s(G_d)$ on the sphere are the same, where $\iota_s(G)$ and $\iota_s(G_d)$ are dual graphs of each other.



(a) F-plane graph (b) V-plane graph

Figure 5.2: F-plane graph and V-plane graph based on the same weaving

Given a weaving σ, a weaving σ' is a ***sub-weaving*** of σ if $\sigma' \subseteq \sigma$ and σ'-

imbedding is a connected component of σ-imbedding.

Construct F-Planar Graph
INPUT: a connected plane weaving σ.
OUTPUT: An ERS $\rho(G)$ of a connected planar graph G.

1 Classify the gaps of the σ into two subsets A_F and A_V, corresponding to the F-gaps and V-gaps of σ, such that the exterior gap r_0 is in A_F.
2 **For** each gap r_j in A_V,
add a star inside the gap r_j, with its center v_j in r_j, and an edge from v_j to each corner of r_j.
3 Remove all edges of ι_σ.
4 **For** each vertex v_i of ι_σ, which corresponds to a cross s_i in σ,
smooth the degree-2 vertex v_i and label the smoothed edge the same twist-type with s_i.

Figure 5.3: Algorithm to construct F-planar graph

Then, we present the main theorem of this section.

Theorem 5.1.3 *Given a disconnected plane weaving σ, there exists an ERS of a planar graph G induces σ.*

PROOF. Let σ be a given plane weaving, and $\sigma_1, \sigma_2, \ldots, \sigma_k$ be sub-weavings of σ. We prove it with induction on the number of components of the σ-imbedding. We name σ as σ^i, if the σ-imbedding contains i components.

According to Theorem 5.1.2, there exists an ERS $\rho(G^1)$ of a planar graph G^1, which induces σ^1.

Assume our statement is also true for a plane weaving σ^{k-1}. The σ^{k-1}-imbedding contains $k-1$ components ($k > 1$), which can be induced from an ERS $\rho(G^{k-1})$ of a planar graph G^{k-1}. Note that G^{k-1} may not have $k-1$ components. Since σ^{k-1} is 2-colorable, we can classify the gaps of σ^{k-1} into two subsets, F-gaps and

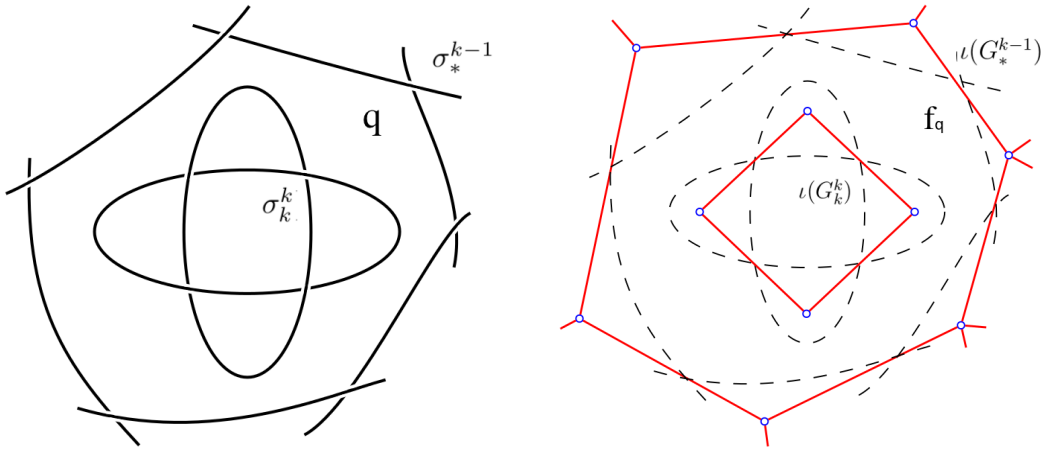
V-gaps, according to the colors, and the unbounded gap is classified as a F-gap. The regions and vertices of $\iota(G^{k-1})$ correspond to F-gaps and V-gaps respectively, and the unbounded region corresponds to the unbounded F-gap.

Then, we consider a plane weaving σ^k, whose σ^k-imbedding contains k components. Assume the σ^k-imbedding $\iota_{\sigma^k} : G_{\sigma^k} \to \mathbb{R}^2$ has connected components $C_1, C_2, \ldots, C_k$, corresponding to sub-weavings $\sigma_1^k, \sigma_2^k, \ldots, \sigma_k^k$ of σ^k, where C_i is the connected planar graph σ_i^k-graph. We construct the tree of plane graph T_k rooted at a virtual graph, with the components $C_1, C_2, \ldots, C_k$ as nodes. Without loss of generality, we assume C_k is a leaf in the tree T_k. Since C_k is a leaf of the tree, there is no sub-weavings inside the inner gaps of σ_k^k. Assume the sub-weaving σ_k^k is entirely inside a gap q, which may be the unbounded gap or an inner gap of another sub-weaving. By removing σ_k^k from σ^k, there are exactly $k-1$ sub-weavings in $\sigma^k - \{\sigma_k^k\}$. According to our hypothesis, there exists an ERS $\rho(G_*^{k-1})$ of a planar graph G_*^{k-1}, such that the induced weaving of $\rho(G_*^{k-1})$ is σ_*^{k-1}, which is topologically equal to $\sigma^k - \{\sigma_k^k\}$, and each gap in σ_*^{k-1} either corresponds to a region or a vertex of $\iota(G_*^{k-1})$. Accordingly, the gap q may correspond to a region or a vertex in $\iota(G_*^{k-1})$.

If q corresponds to a region, we assume f_q is the corresponding region in $\iota(G_*^{k-1})$. Since C_k is a connected plane graph, these exists an ERS $\rho(G_k^k)$ of a F-planar graph G_k^k to induce σ_k^k, whose underlying imbedding is C_k. Let the imbedding of G_k^k be $\iota(G_k^k)$ with the pure rotation system $\rho^0(G_k^k)$. We create the imbedded graph $\iota(G_*^k)$ by imbedding $\iota(G_k^k)$ disjointly inside the region f_q of $\iota(G_*^{k-1})$. The ERS $\rho(G_*^k)$ is the disjoint union of $\rho(G_*^{k-1})$ and $\rho(G_k^k)$, as shown in Figure 5.4. Since the disjoint union of plane graphs is a plane graph, the imbedded graph $\iota(G_*^k)$ is a plane graph.

Let the induced weaving of $\iota(G_*^k)$ be σ_*^k, and q' be the corresponding gap of f_q. Then we show the weaving σ_*^k is topologically equal to σ^k. We construct the

tree of plane graph T'_k for σ^k_*-imbedding, and we show the tree of plane graphs T'_k is isomorphic to T_k, and each component of T'_k is plane-isomorphic to the image component of T_k. We already know that σ^{k-1}_* is topologically equal to $\sigma^k - \{\sigma^k_k\}$. Hence, the tree of plane graph $T'_k - \{C'_k\}$ is isomorphic to $T_k - \{C_k\}$, and each pair of corresponding components are plane-isomorphic. Since C_k is plane-isomorphic to C'_k, we only need to check whether T_k and T'_k is isomorphic or not. Considering σ^k_k is inside the gap q, and σ'_k is inside the corresponding gap q', T_k is isomorphic to T'_k.



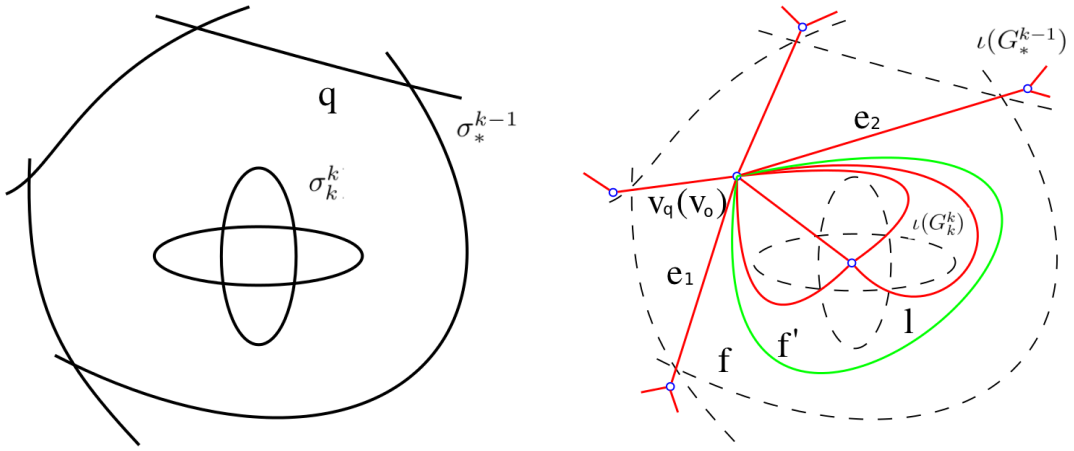
(a) Weaving σ^k (b) Plane graph $\iota(G^k)$

Figure 5.4: The gap q corresponds to a region of $\iota(G^{k-1}_*)$

If q corresponds to a vertex, we assume v_q is the corresponding vertex to q in $\iota(G^{k-1}_*)$. We construct an ERS $\rho(G'^k_k)$ of a V-planar graph G'^k_k, where the exterior gap of σ^k_k is corresponding to the exterior vertex v_o of $\iota(G'^k_k)$. Let (v_q, e_1, e_2) be a face-corner at the vertex v_q in $\iota(G^{k-1}_*)$. We add an untwisted self-loop l at the vertex v_q inside an incident face f, such that the self-loop l starts and ends in the same face corner (v_q, e_1, e_2). The self-loop l creates a new face f' inside the self-loop. We imbed

$\iota(G_k'^k)$ inside the face f' by identifying the exterior vertex v_o with the vertex v_q, such that $\iota(G_k'^k)$ and $\iota(G_*^{k-1})$ only meet at the vertex $v_q(v_o)$. The ERS $\rho(G_*^k)$ is the union of $\rho(G_*^{k-1})$, $\rho(G_k'^k)$ and an untwisted self-loop l at the face corner (v_q, e_1, e_2), as shown in Figure 5.5. Since no edge intersection is introduced in the operations, the graph G_*^k is a planar graph.

Let the induced weaving of $\rho(G_*^k)$ be σ_*^k. Since the self-loop l is untwisted, by applying the Strand-Tracing Algorithm, the two strands along the self-loop separates the weaving σ_*^k into two parts, one is inside the self-loop, called $\sigma_*^k(1)$, and the other weaving is called $\sigma_*^k(k-1)$. It is trivial to realize the $\sigma_*^k(k-1)$ is topologically equal to σ_*^{k-1}. Then we consider the other weaving $\sigma_*^k(1)$.



(a) Weaving σ^k (b) Plane graph $\iota(G^k)$

Figure 5.5: The gap q corresponds to a vertex of $\iota(G_*^{k-1})$

Let (v_o, e_2', e_1') be the face-corner of the unbounded gap in $\iota(G_k'^k)$. After inserting the self-loop l at (v_o, e_2', e_1'), the inner strand of the self-loop l is merged into the strand $(c_{e_1'}, c_{e_2'})$ in $\sigma_*^k(1)$, where $c_{e_1'}$ and $c_{e_2'}$ are the crosses corresponding to the two

edges e'_1, e'_2. Let the $\sigma^k_*(1)$-imbedding be C''_k. We show C''_k is plane-isomorphic to C_k. Let the gap corresponding to v_q be q'' in σ^k_*. Since the self-loop l is at the vertex v_q, the gap q'' is incident to the two strands of l, and considering the inner strand of l is merged into (c_{e_1}, c_{e_2}), the gap q'' is also incident to the strand (c_{e_1}, c_{e_2}). By removing the weaving $\sigma^k_*(k-1)$, the gap q'' is the unbounded gap. Therefore, the exterior gap of C''_k is corresponding to the exterior gap of C_k. Considering the pure rotation systems of C''_k and C_k is equal, C''_k are plane-isomorphic to C_k.

Therefore, for a plane weaving σ^k, whose σ^k-imbedding contains k components, there exists an ERS $\rho(G^k)$ of G^k is the weaving σ^k. □

Figure 5.6 gives an algorithm to constrcut a planar graph from a plane weaving.

General Planar Graph
INPUT: a plane weaving σ.
OUTPUT: an ERS ρ_G of a planar graph G whose induced weaving is σ.

1 Construct the tree of plane graph T_σ with the components $C_1, C_2, \ldots, C_r$ of σ-imbedding.
2 Generate an queue Q of the tree nodes by the breadth-first search algorithm, and remove the virtual root from Q.
3 Create an empty imbedded graph $\iota(G)$.
4 **while** the queue Q is not empty,
4.1 Let C_i be the first component in Q.
4.2 **if** C_i is a child of the root,
imbed F-planar graph $\iota_F(C_i)$ in the unbounded gap of $\iota(G)$ disjointly.
4.3 **else if** C_i is in a gap p, corresponding to a F-gap, of its parent C_j in T_σ,
imbed F-planar graph $\iota_F(C_i)$ inside the corresponding face-gap f_p of $\iota(C_j)$ disjointly.
4.4 **else if** C_i is in a gap p, corresponding to a V-gap, of its parent C_j in T_σ,
4.4.1 add a self-loop l inside a face, where l begins and ends at the same face-corner of v_p in $\iota(C_j)$.
4.4.2 imbed the V-planar graph $\iota_V(C_i)$ inside l, and attach $\iota_V(C_i)$ to v_p by identifying the outer vertex v_{o_i} of $\iota_V(C_i)$ with v_p.
4.5 Remove C_i from Q.
5 The ERS ρ_G is the union of ERS of $\iota(C_i)$ plus the untwisted inserted self-loops.

Figure 5.6: Algorithm to construct general planar graph

5.2 Weaving on Torus

Lemma 5.2.1 *Let σ be a connected weaving on a torus S_1. There is at most one non-cellular gap in σ.*

PROOF. Assume the weaving σ is not a cellular weaving. Let p be a non-cellular gap in σ. There exits a non-contractable cycle c in the gap p according to the Poincaré Conjecture. There are two possibilities for the cycle c, either c is a separating cycle or a non-separating cycle on S_1.

If c is a separating cycle on S_1, cutting along c and patch the two open discs to the two holes would separating S_1 into two parts, a torus S_1' and a sphere S'. Since our cut does not touch the weaving, the weaving should be still connected. Thus, the weaving σ should be either on the torus S_1' or on the sphere S'. The weaving σ can not be on S_1'. Otherwise, the gap p is topologically equivalent to a sphere, and the cycle c is contractable in the gap p. It contradicts with our assumption. Thus, the weaving σ is on the sphere S_0'. A connected weaving on a sphere is a cellular weaving, and no other gap in σ is touched in our operation. The gap p is the only non-cellular gap in σ.

If c is a non-separating cycle on S_1, cutting along c and patch the two open discs reduce the torus S_1 into a sphere S_0'. Similarly, the operation dose not touch other gaps and strands of σ. The weaving on the sphere S_0' is a cellular weaving. The gap p is the only non-cellular gap in σ. □

Corollary 5.2.2 *A connected and 2-colorable weaving σ on a torus is a normal weaving.*

Lemma 5.2.3 *If a weaving σ is a normal weaving, each sub-weaving of σ is normal.*

PROOF. We prove it with induction. If the normal weaving σ^1 contains only one sub-weaving. It is trivial to realize the sub-weaving is normal. Now we assume a normal weaving with $k-1$ sub-weavings, the claim is still true, i.e. each sub-weaving is normal. Then we consider a normal weaving σ^k with k sub-weavings. Assume σ_0 is a sub-weaving of a normal weaving σ^k. Let the other sub-weavings in σ^k be $\sigma_1, \sigma_2, \ldots, \sigma_{k-1}$. It is easy to realize that a gap whose boundaries contain strands from both σ_0 and σ_i, $1 \leq i \leq k-1$, is non-cellular with disjoint gap boundaries. Let $g_1, g_2, \ldots, g_r$ be such gaps in σ^k whose boundaries contain strands from both σ_0 and σ_i. Then the gaps $g_1, g_2, \ldots, g_r$ are with the same color in σ^k, since these gaps are non-cellular and σ^k is normal. By removing the sub-weaving σ_0 from σ^k, the gaps $g_1, g_2, \ldots, g_r$ are merged as one gap g_0, whose color is the same with $g_1, g_2, \ldots, g_r$. The gap g_0 can be either cellular or non-cellular, which would not affect that the new obtained weaving σ'_{k-1} is normal. Thus, according to our assumption, the sub-weavings $\sigma_1, \ldots, \sigma_{k-1}$ are normal. Similarly, by removing the sub-weaving σ_1, we can also show $\sigma_0, \sigma_2, \ldots, \sigma_{k-1}$ are normal. Thus, all sub-weaving of σ^k are normal. □

Corollary 5.2.4 *If a weaving σ is induced by an ERS of a graph on a torus, each sub-weaving of σ is normal.*

We say an imbedded graph $\iota(G)$ is ***cellular-based*** on a surface S, if one of its components C is cellular imbedded on S, where the other components are disjointly contained in cellular faces. The cellularly imbedded component C defines the surface S, and it is much more meaningful than the disconnected imbedded graph with no cellular components. Therefore, we only discuss the weaving induced from an ERS of a cellular-based imbedded graph.

Theorem 5.2.5 *Given a weaving σ immersed on a torus S_1, where each sub-weaving*

of σ is 2-colorable, there exists an ERS $\rho(G)$ of a graph G induces σ, where $\iota(G)$ is cellular-based.

PROOF. We prove it by induction on the number of sub-weavings in σ. We name σ as σ^i, if σ contains i sub-weavings. According to Corollary 5.2.2, σ^1 is a normal weaving, and Theorem 4.2.3 indicates that there exists an ERS $\rho(G^1)$ of a graph G^1 that induces σ^1, where the imbedded graph $\iota(G^1)$ is cellular. We assume that a weaving σ^{k-1}, whose sub-weavings are 2-colorable, is inducible from an ERS $\rho(G^{k-1})$ of a graph G^{k-1} where $\iota(G^{k-1})$ is cellular-based. Then we consider a weaving σ^k on a torus with 2-colorable sub-weavings.

Since there are more than one sub-weaving in σ^k $(k > 1)$, there exists at least one non-cellular gap in σ^k. We pick a non-cellular gap q, where we can find a non-contractable cycle c in q. The cycle c is either a non-separating cycle or a separating cycle on the torus S_1. We cut along c, and patch the two holes with two open discs d_1, d_2. The non-cellular gap q is separated into two gaps q_1, q_2, where q_1, q_2 can be the same gap. The open discs d_1, d_2 are inside the two gaps q_1, q_2 respectively, and d_1, d_2 are disjoint if q_1, q_2 are the same. Let the weaving obtained after cutting be σ^k_*.

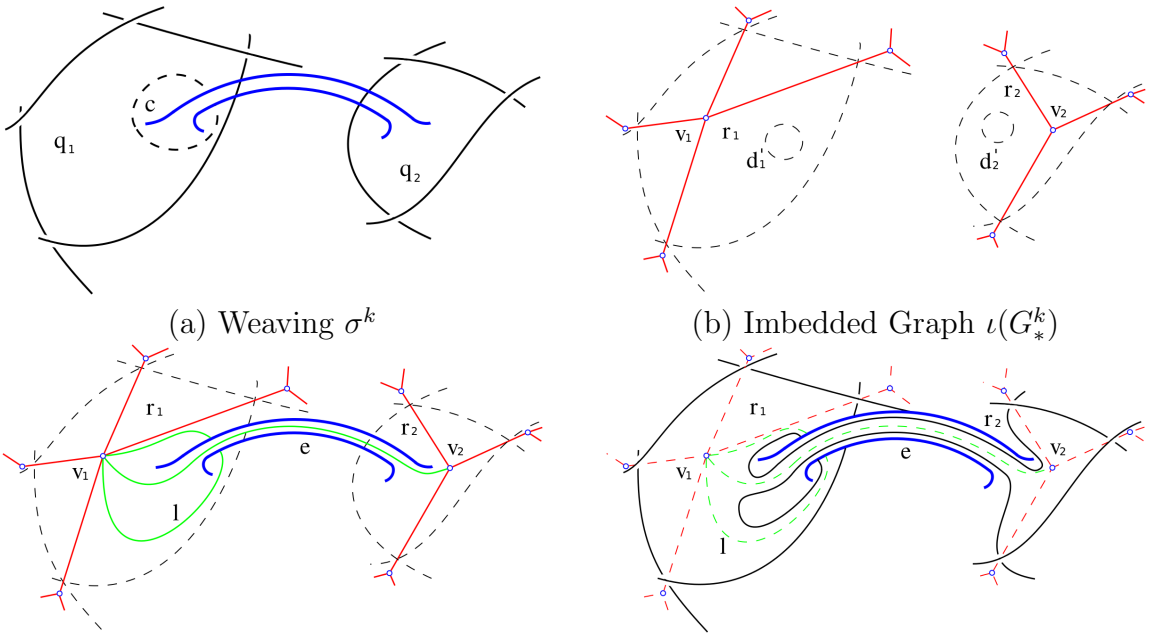
If the cycle c is a non-separating cycle on S_1, after cutting c and patching the open discs d_1, d_2, the obtained surface is a sphere S_0. Since the cycle c is entirely inside the gap q of σ^k, the cut-and-patch operation dose not affect the connectivity of a weaving. There are still k sub-weavings in σ^k_*. We already know that a connected weaving on a sphere is 2-colorable. Each sub-weaving of σ^k_* is 2-colorable, and σ^k_* is also 2-colorable. We claim that q_1, q_2 can be colored with the same color. Otherwise, if we reverse the cutting operation by removing the open discs d_1, d_2 and adding a handle between them to merge q_1, q_2, the restored σ^k is not 2-colorable. There are

two possibilities for q_1, q_2, such as

1. If the gaps q_1, q_2 correspond to two vertices v_1, v_2 of $\iota(G^k_*)$, as shown in Figure 5.7, we take two regions r_1, r_2 of $\iota(G^k_*)$ which are incident to v_1, v_2 respectively. Let d'_1, d'_2 be two open discs inside the regions r_1, r_2. If r_1, r_2 are the same region, d'_1, d'_2 should be disjoint. We add a self-loop l at v_1 in the gap r_1, such that the self-loop l bounds the open disc d'_1 inside the face which is only defined by l. We remove the open disc d'_1, d'_2 and add a handle between them. Then we add an edge e to connect v_1, v_2 through the handle without intersecting any other edge in $\iota(G^k_*)$. By labeling the edges l and e as untwisted, we claim the ERS $\rho(G^k_{**})$ of G^k_{**} induces the given weaving σ^k. Since v_1, v_2 are connected with an untwisted edge e, the corresponding gaps of v_1, v_2 in the induced weaving σ^k_{**} are the same gap q'. The untwisted self-loop l keeps the handle inside the gap q' and the other edges unaffected. The obtained weaving is the same with the weaving by removing the open disc d_1, d_2 in σ^k_* and adding handle between them. The induced weaving σ^k_{**} is σ^k.

   Then we show the obtained imbedded graph $\iota(G^k_{**})$ is cellular-based on a tours. Considering each connected planar graph imbedded on a sphere is cellular, the plane graph $\iota(G^k_*)$ is cellular-based. Let the components of G^k_* be $C_1, C_2, \ldots, C_t$ $(0 \leq t \leq k)$. We assume the vertices v_1, v_2 are in the components C_i and C_j, where C_i, C_j may be the same component. After adding edge e to connect v_1, v_2, the two components C_i, C_j are connected as one component C. The added self-loop l and the edge e make the component C is cellular on the torus S_1, if we ignore the other components. The obtained $\iota(G^k_{**})$ is cellular-based.

2. If the gaps q_1, q_2 correspond to two regions r_1, r_2 of $\iota(G^k_*)$, we do the "dual" graph of $\iota(G^k_*)$ and let q_1, q_2 correspond to two vertices in the "dual" graph to



(a) Weaving σ^k

(b) Imbedded Graph $\iota(G_*^k)$

(c) Imbedded graph $\iota(G_{**}^k)$

(d) Induced weaving σ^k

Figure 5.7: If the two gaps q_1, q_2 correspond to two vertices in $\iota(G_*^k)$

apply the operation illustrated above. Since σ_*^k is immersed on a sphere, each sub-weaving of σ_*^k is a cellular weaving if we remove the other sub-weavings of σ_*^k. For a connected graph in the sphere, the simply-twisted ERS of the dual graph can induce the same cellular weaving with simply-twisted ERS of the original graph. Thus, we construct the "dual" graph of $\iota(G_*^k)$ as follows. We pick a weaving σ_1 which is incident to the gap q_1. By removing the other sub-weavings, we can obtain an ERS $\rho(G_*^1)$ of planar graph G_*^1, which induces the sub-weaving σ_1 and the gap q_1 corresponds to a vertex v_1 in G_*^1. Then, similarly with the proof in Theorem 5.1.3, we create a tree of plane graph rooted at σ_1-imbedding and inductively consider the other sub-weavings in the tree. We can obtain a simply-twisted ERS of a graph $G_*^{\prime k}$ which induces σ_*^k and q_1, q_2 are corresponds to two vertices v_1, v_2 of $G_*^{\prime k}$.

If the cycle c is a separating cycle on the torus S_1, the torus is separated into a torus S_1' and a sphere S_0'. The number of sub-weavings on the torus S_1' is less than k, otherwise c is not a non-contractable cycle. Assume there are h sub-weavings on S_1', where $0 \leq h < k$. We call the weaving on the torus as σ_*^h, and the weaving on the sphere as σ_*^{k-h}. It is possible that h is equal to zero. Since c is a separating cycle, it is safe to assume the two open discs d_1, d_2 are in the torus S_1' and the sphere S_0' respectively. Similarly, the two gaps q_1, q_2 are in the two surface S_1', S_0' respectively. According to our assumption, there exists an ERS $\rho(G_*^h)$ of a graph G_*^h induces the weaving σ_*^h, where $\iota(G_*^h)$ is cellular-based. The gap q_1 is corresponding to a vertex v_1 or a gap r_1 of $\iota(G_*^h)$. We discuss the two possibilities as follows.

1. If the gap q_1 corresponds to a vertex v_1, we construct a simply-twisted ERS $\rho(G_*^{k-h})$ of a planar graph G_*^{k-h} on the sphere S_0', such that the gap q_2 is corresponding to a vertex v_2 in G_*^{k-h}. Since d_1, d_2 are inside the gaps r_1, r_2 respectively, which are incident to the vertices v_1, v_2. Similarly, we insert a self-loop l at v_1 to bound d_1 inside the face only defined by l, remove the open discs d_1, d_2, and add a handle to connected them. Then, we add an edge e to connect v_1, v_2 without intersect with the other edges. The edge e and the self-loop l are untwisted. Similarly, the obtained imbedded graph $\iota(G_{**}^k)$, combining the two imbedded graph $\iota(G_*^h)$ and $\iota(G_*^{k-h})$ with edges l and e, is cellular-based, and the induced weaving is σ^k.

2. If the gap q_1 corresponds to a region r_1, we construct a simply-twisted ERS $\rho(G_*^{k-h})$ of a planar graph G_*^{k-h} on the sphere S_0', such that the gap q_2 is corresponding to a region r_2 in G_*^{k-h}. We remove the open discs d_1, d_2 and add a handle to connected them. Thus, the torus S_1' and the sphere S_0' are combined into a torus S_1''. It is easy to realize the induced weaving of the obtained ERS

$\rho(G^k_{**})$ is exactly σ^k by reversing the cut-and-patch operation. Since $\iota(G^h_*)$ is cellular-based, the obtained imbedding $\iota(G^k_{**})$ is also cellular-based.

This completes the proof. □

Corollary 5.2.6 *A weaving σ on a torus is inducible from an ERS $\rho(G)$ of a graph G, where $\iota(G)$ is cellular-based if and only if each sub-weaving of σ is 2-colorable.*

In this chapter, we take the advantages of the weaving structures in low genus surfaces and break the limitations for normal weavings. We point out that any connected weaving is normal on a sphere and has at most one non-cellular gap on a torus. Moreover, we introduce a hierarchy structure to describe the disconnected weaving structures on a sphere and a torus. This hierarchy structure helps us to construct the extended rotation system of a "semi-structured" graph from any weaving on a sphere and 2-colorable weavings on a torus, which permits us to induce most weaving artworks in practice, like Celtic knots.

6. WEAVING DOUBLING OPERATIONS AND IMBEDDINGS SUBDIVISION OPERATIONS

In this chapter, we demonstrate how extended rotation systems of graphs provide a practical model for weavings on general topological surfaces. In particular, we establish a correspondence between two operations on weavings that are well-established in the computer graphics community, the Catmull-Clark subdivision and the Doo-Sabin subdivision, with topological operations on extended rotation systems of a graph. Under this straightforward correspondence, these two operations can be readily implemented by a graphics system based on extended rotation systems. For describing the correspondence with the Catmull-Clark operation, we introduce a new *doubling operation* on weavings.

6.1 Catmull-Clark Subdivisions

Given a cellular imbedding $\iota : G \to S_i$ of a graph G, as in Figure 6.1(a), with induced pure rotation system ρ_0, the ***Catmull-Clark***[1] ***subdivision operation*** [10] builds a new graph G^{cc} and its imbedding $\iota^{cc} : G^{cc} \to S_i$ in that same surface, with induced pure rotations system ρ_0^{cc} as follows:

(1) into the interior of each edge $e \in E_G$, insert a vertex w_e, as in Figure 6.1(b);

(2) into the interior of each face f in $\rho_0(G)$, insert a vertex w_f, as in Figure 6.1(c);

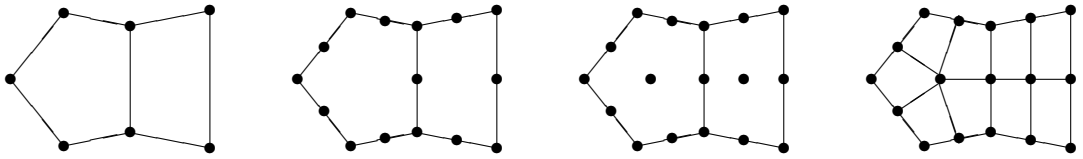
(3) for each face f and each edge e on the boundary of the face f in $\rho_0(G)$, add an edge $[w_f, w_e]$, as in Figure 6.1(d), which also depicts the induced pure rotation system on G^{cc};

[1]We note that the Catmull-Clark algorithm is used in computer graphics to create smooth subdivided surfaces. In that context, the geometric locations of the new (and old) vertices are carefully considered. Our present discussion focuses on the topological properties of the resulting imbedding.

(4) if the oriented fb-walk for face f is $d_1, d_2, \ldots, d_m$, then the induced rotation at the vertex w_f is w_f. $[w_f, w_{e_1}], [w_f, w_{e_2}], \ldots, [w_f, w_{e_m}]$; and

(5) if edge $e = [u, v]$ is incident on faces f and f', and if the corner (u, x, e) of face f immediately precedes the corner (v, e, y), as the fb-walk traverses edge e, then the induced rotation at the vertex w_e is w_e. $[w_e, u], [w_e, w_f], [w_e, v], [w_e, w_{f'}]$.

Topologists will recognize that the graph G^{cc} is a subgraph of the 1-skeleton of the barycentric subdivision of the 2-complex specified by the imbedding $\iota : G \to S_i$.



(a) imbedding $\iota : G \to S_i$ (b) add edge-vertices (c) add face-vertices (d) add new edges

Figure 6.1: Catmull-Clark subdivision

6.2 The Doubling Operation on Weavings

We now examine how a Catmull-Clark subdivision is realized in an ERS of a graph and a consequence on the induced weaving. For this, we first introduce a new operation on weaving.

Definition Let c be a strand in a weaving σ in a surface S_i. We ***double the strand*** c in the weaving σ as follows:

(1) we replace strand c by two parallel and closely positioned strands c_1 and c_2, so that the only crossings of σ that are affected are those on the strand c;

(2) each crossing of strand c with itself or with any other strand c' is replaced by two consecutive crossings on c or c', one with c_1 and the other with c_2;

(3) the crossing-types of the new crossings are left undetermined for the time being.

Definition Let σ be a weaving on a surface S_i. The result of doubling every strand is called a ***doubling of the weaving*** σ^2. (We will focus on doublings that are alternating weavings.)

Figure 6.2 shows a twill and a possible doubling and redoubling, in which crossing-types were assigned so that the new weavings are twills. In a twill, each strand goes two over and two under, cyclically, with one shift between two adjacent strands.

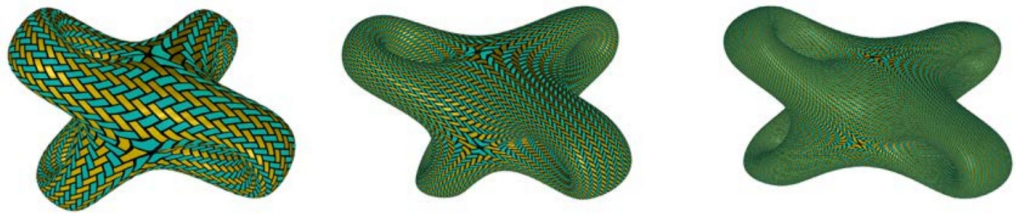


Figure 6.2: A weaving, doubled and redoubled.

Doubling a weaving creates a "refined" weaving on the same surface. Further importance of doubling will emerge in next subsections and chapters, where we see that the assignment of crossing types will allow us to construct different weaving patterns on a topological surface. An immediate indication of the utility of doubling is revealed by the following theorem.

Theorem 6.2.1 *Let σ be any weaving. Then a doubled weaving σ^2 is a normal weaving.*

PROOF. We recall that a weaving is normal if its gaps can be colored with two colors such that no two adjacent gaps are assigned with the same color and that all gaps in one of the colors are cellular. In a doubled weaving, every gap that corresponds to a crossing or face-gap of the original weaving can be colored blue, and every gap that corresponds to a former strand segment (from crossing to crossing) can be colored red, as illustrated in Figure 6.3. In this proper 2-coloring, every red region is cellular. □

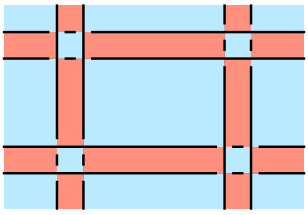


Figure 6.3: 2-coloring the gaps of a doubled weaving.

Remark 6.2.2 *We observe that Theorem 7.2.9 holds true for any weaving σ on any surface. In particular, the weaving σ need not be inducible by an ERS of a graph.*

The following theorem reveals how a Catmull-Clark subdivision can be realized by the weaving doubling operation.

Theorem 6.2.3 *Let ρ be a simply-twisted ERS of a connected graph G with pure rotation system ρ_0 and induced weaving σ, and let ρ^{cc} be any simply-twisted ERS of the graph G^{cc} whose pure rotation system ρ_0^{cc} is obtained by applying Catmull-Clark subdivision to ρ_0. Then the weaving σ^{cc} induced by ρ^{cc} is obtainable by assigning appropriate crossing-types to a doubling of the weaving σ.*

PROOF. Figure 6.4 serves as an intuitive guide through the combinatorial details of the proof. The area of chief concern is highlighted in yellow.

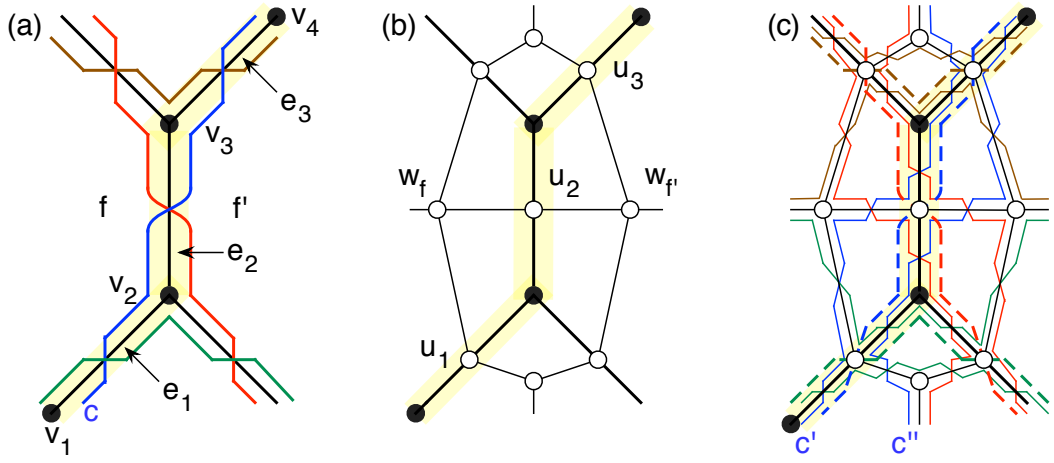


Figure 6.4: (a) The weaving strands induced by a simply-twisted ERS. (b) The Catmull-Clark subdivision of the imbedded graph that induces the weaving. (c) Doubling the strands.

Since the ERS ρ is simply-twisted, each edge of the graph G induces exactly one crossing, which we place at the middle of the edge. Consider three consecutive crossings along a strand c (shown in blue) of the weaving σ, say along three edges $e_1 = [v_1, v_2]$, $e_2 = [v_2, v_3]$, and $e_3 = [v_3, v_4]$, respectively. Therefore, in the pure rotation system ρ_0, the triple (v_2, e_1, e_2) makes a corner of a face f, the triple (v_3, e_2, e_3) makes a face corner of a face f', and edge e_2 is on the boundary of the faces f and f'. Figure 6.4 assumes, without loss of generality, that the strand c is on the right side of edge e_1 before it traverses that edge e_1 (along the direction of traversal).

Applying Catmull-Clark subdivision to the pure rotation system ρ_0 yields the pure rotation system ρ_0^{cc}, as shown in Figure 6.4(b), where new vertices w_f and $w_{f'}$

are inserted into faces f and f', respectively; new vertices u_1, u_2, and u_3 are inserted at the middle of the edges e_1, e_2, and e_3, respectively; and new edges $[w_f, u_1]$, $[w_f, u_2]$, $[w_{f'}, u_2]$, and $[w_{f'}, u_3]$ are added. The new vertices are shown as hollow circles and the new edges as thin line segments. To avoid cluttering the diagram, we have not transported labels from Figure 6.4(a).

Now consider a strand c' (thin blue) that starts at the right side of the edge $[v_1, u_1]$ in the weaving σ^{cc} induced by the ERS ρ^{cc}. Since the ERS ρ'^{cc} is also simply-twisted, it is easy to verify, as shown in Figure 6.4(c), that the order in which the strand c' crosses edges of the subdivided graph G^{cc} is

$$[v_1, u_1], \quad [u_1, w_f], \quad [w_f, u_2], \quad [u_2, v_3], \quad [v_3, u_3]$$

Observe that this sequence of segments of the strand c' can be drawn in parallel with the part of the strand c (broken blue line) between the crossings on edges $[v_1, v_2]$ and $[v_3, v_4]$. Indeed, the entire strand c' can be drawn "in parallel" with the strand c on the surface, in the sense of deforming strand c' to strand c without passing through any edges of the initial graph G. This is because the crossing of c' on the edge $[v_3, u_3]$ repeats the configuration of its crossing on the edge $[v_1, u_1]$, relative to its relationship with strand c; that is, it starts at the right side of the edge $[v_3, u_3]$. Thus, just as the strand c' lies parallel to the strand c in their course along edges e_1 and e_2, it will also lie parallel to the strand c in their course along edge e_3 and whatever edge strand c traverses immediately after edge e_3. This parallel course continues over the entire course of strands c and c', until they both return to vertex v_1. (The strand c has an even number of crossings because the surface of the weaving is orientable.)

Similarly, if we traverse the part of the strand c'' in the weaving σ^{cc} that starts at vertex u_1 from the right side of the edge $[u_1, v_2]$ in the imbedding of G^{cc}, we can show

that the strand c'' can also be drawn in parallel with the strand c on the surface. Thus, the two parallel strands c' and c'' in the weaving σ^{cc} correspond to the result of doubling the strand c in the weaving σ. That there no strands other that c' and c'' in the weaving σ' follows from these simple facts:

1. the number of edges in the graph G^{cc} is exactly four times that in the graph G (see the edge $[v_2, v_3]$ in the middle figure of Figure 6.4);
2. the number of crossings in the weaving σ^{cc} is exactly four times that in the weaving σ (one on each edge of G^{cc}); and
3. each edge in a simply-twisted extended rotation system induces exactly one crossing.

This completes the proof that the weaving σ^{cc} can obtained by doubling the weaving σ. □

6.3 Doo-Sabin Subdivisions

The Doo-Sabin[2] subdivision algorithm [16] was invented around the same time as the Catmull-Clark algorithm.

Given a cellular imbedding $\iota : G \to S_i$, as in Figure 6.5(a), the Doo-Sabin operation on a graph imbedding $G \to S_i$ constructs a new graph G^{ds} and its imbedding $\iota^{ds} : G^{ds} \to S_i$ in the same imbedding surface as follows:

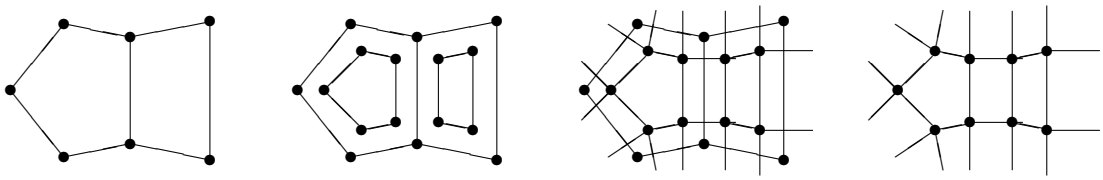
(1) into the interior of each polygonal face f, we insert a cycle of the same length as the fb-walk of f, near to the fb-walk, as in Figure 6.5(b);

(2) for each edge $e \in E_G$, join corresponding endpoints of the two parallel edges in the faces incident on e, as in Figure 6.5(c); and

[2]Like the Catmull-Clark algorithm, the Doo-Sabin algorithm is used is computer graphics to create smooth subdivided surfaces.

(3) discard the original graph G, as in Figure 6.5(d).

Topologists will recognize that the graph G^{cc} is a subgraph of the 1-skeleton of the barycentric subdivision of the 2-complex specified by the imbedding $\iota : G \to S_i$.



(a) imbedding $\rho_0(G)$ (b) add inner-polygons (c) add inner-edges (d) remove original graph G

Figure 6.5: Doo-Sabin subdivision of a graph imbedding $G \to S_i$

6.4 Dual Imbeddings

A pure rotation system ρ_0 on a connected graph G corresponds to a cellular imbedding $\iota : G \to S_i$ into an orientable surface. We recall that a ***dual imbedding*** $\iota^* : G^* \to S_i$ can be constructed as follows:

(1) install a dual vertex w_f into the interior of each face f of the imbedding; and

(2) through each edge e of the primal graph G, install an edge e^* that crosses edge e and joins the dual vertex w_f in the face on one side of e to the dual vertex $w_{f'}$ in the face on the other side of e. (Faces f and f' may be the same face).

The dual imbedding $\iota^* : G^* \to S_i$ induces a ***dual pure rotation system*** ρ_0^* on the dual graph G^*.

Theorem 6.4.1 *Let ρ and ρ^* be two simply-twisted extended rotation systems on graphs G and G^*, respectively, whose pure rotation systems ρ_0 and ρ_0^* are dual to each other. Let the edge-twisting value assigned to each dual edge $e^* \in E_{G^*}$ be*

opposite to the edge-twisting value assigned to the corresponding primal edge $e \in E_G$. Then the weavings σ and σ^* induced by ρ and ρ^*, respectively, are isomorphic.

PROOF. We recall from Proposition 4.1.5 that the graph of the weaving induced by a simply-twisted extended rotation system is equivalent to the imbedding of the medial graph for the imbedding induced by the corresponding pure rotation system. Since a graph and its dual have the same medial graph imbedding, the theorem follows. □

The following theorem can be easily verified. The duality is illustrated by Figure 6.6.

Theorem 6.4.2 *Let $\iota : G \to S_i$ be an imbedding of a connected graph G. Let $\iota^{cc} : G^{cc} \to S_i$ be the imbedding obtained by applying Catmull-Clark subdivision to $\iota : G \to S_i$, and let $\iota^{ds} : G^{ds} \to S_i$ be the imbedding obtained by applying Doo-Sabin subdivision to $\iota : G \to S_i$. Then $\iota^{cc} : G^{cc} \to S_i$ and $\iota^{ds} : G^{ds} \to S_i$ are dual imbeddings.*

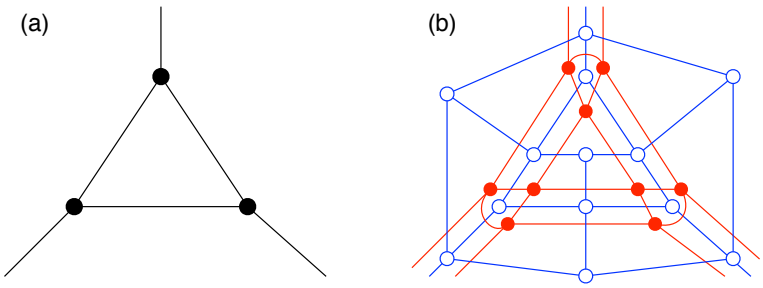


Figure 6.6: (a) A graph imbedding. (b) Superposition of the Clark-Catmull (blue) and Doo-Sabin (red) subdivisions.

Combining Theorem 6.4.1 and Theorem 6.4.2, and using Proposition 4.1.5, we obtain our final theorem:

Theorem 6.4.3 *Let ρ be a simply-twisted ERS of a connected graph G with pure rotation system ρ_0, and let ρ^{ds} be a simply-twisted ERS of a graph G^{ds} whose pure rotation system ρ_0^{ds} is obtained from ρ_0 by applying Doo-Sabin subdivision. Then the weaving σ^{ds} induced by ρ^{ds} is obtainable by doubling the weaving σ induced by ρ and then assigning appropriate edge-twistings.*

Theorem 6.4.2 and Theorem 6.4.3 present the relationships between doubling operation and Catmull-Clark and Doo-Sabin subdivisions. Therefore, doubling operation could be a standard surgery operation on a weaving with the benefits from the two classical subdivision operations. By applying the doubling operation, we can refine the appearance a given weaving. In the next chapter, we would like to explore more properties of this surgery operation on a weaving.

7. ALTERNATING WEAVINGS, TWILLABLE WEAVINGS AND CHILD-TWILLABLE WEAVINGS

In this chapter, we focus on the appearances of weavings. As discussed in Chapter 1, a plain weaving has a checkerboard-like appearance, and a twill weaving has a beautiful diagonal pattern. By changing the twisted-types of the edges of an imbedding, the appearances of the induced weavings could be different. In the following, we concentrate on graph imbeddings and the corresponding ERSs to discuss several classical weavings appearances.

7.1 Alternating Weaving and Twillable Weaving

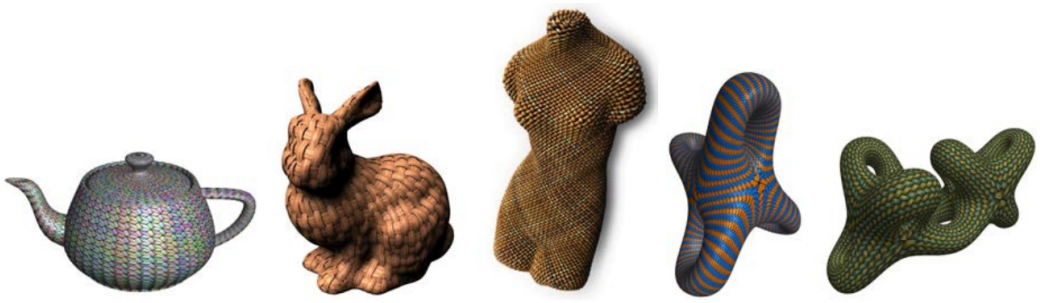


Figure 7.1: Examples of weavings on orientable surfaces

An ***alternating weaving*** is a weaving $\sigma : \mathcal{C} \to S_i$ such that, when traversing the image $\sigma(c)$ of any circuit $c \in \mathcal{C}$, one alternatingly crosses *over* and *under*. Alternating weaving, which is also called ***plain weaving*** in weaving literature, has been among the most widely used weaving patterns in practice [3]. The first three objects in Figure 7.1 are examples of alternating weaving.

An edge in an ERS ρ of a graph G is ***positively twisted*** (resp. ***negatively twisted***) if it is k^+-twisted (resp. k^--twisted) for some integer $k > 0$. An ERS of a graph is ***uni-direction-twisted*** if either none of its edges are positively twisted or none of its edges are negatively twisted. Theorem 7.1.1 is known. For completeness, we give a proof here, which is also useful for our proof of the theorem immediately after.

Theorem 7.1.1 [3] *Let ρ be a uni-direction-twisted ERS of a graph G. Then ρ induces an alternating weaving.*

PROOF. The corresponding pure rotation system ρ_0 induces a cellular imbedding $G \to S_i$ for some genus i. We shall suppose that the ERS ρ is positively twisted. By Lemma 4.1.2, we can assume without loss of generality that each edge in $\rho(G)$ is either 0-twisted or 1^+-twisted. Let σ be the weaving on the surface S_i that is induced by ρ. If the weaving σ contains no crossings, then, by definition, σ is an alternating weaving.

Thus, suppose that the weaving σ has crossings, and suppose that the strand $s = \sigma(c)$ goes "under" at a crossing x. Then the crossing x is induced by a 1^+-twisted edge e in $\rho(G)$. Accordingly, as one traverses the edge e on the surface S_i, the strand s must start at one's right, pass through the crossing x, and then continue on one's left along the edge (see Figure 3.1(b)). After the crossing x, the strand s continues its traversal along an fb-walk of the pure rotation system $\rho_0(G)$, while staying to one's left on each traversed edge, until it encounters the next crossing x'. Let the crossing x' be induced by a 1^+-twisted edge e' (note that x' could be x and that e' could be e). At the crossing x', the strand s then crosses from the left side of the edge e' to the right side. Again because e' is 1^+-twisted, the strand s must go

"over" at the crossing x', as shown in Figure 7.2, where the strand s is colored blue.

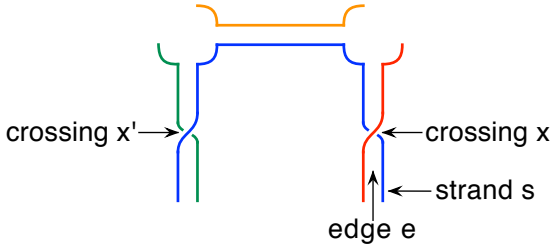


Figure 7.2: The alternating weaving induced by a uni-direction-twisted rotation system.

Similarly, if the strand s emerges from a crossing where it goes "over", then the strand must go "under" at the next crossing. This proves that the weaving σ induced by $\rho(G)$ is an alternating weaving. The case where no edges in $\rho(G)$ are positively twisted can be proved similarly. □

We recall that the σ-graph uses the crossings of σ as vertices and the strand segments in σ as edges. For a normal weaving σ with a connected σ-graph, Theorem 7.1.1 has a converse.

Theorem 7.1.2 *Let $\sigma : \mathcal{C} \to S_i$ be a normal weaving with a connected σ-graph. Then σ is an alternating weaving if and only if it is induced by a uni-direction twisted ERS of a graph.*

PROOF. If the weaving σ is induced by a uni-direction-twisted ERS of a graph, then by Theorem 7.1.1, σ is an alternating weaving.

Now suppose, conversely, that σ is a normal, alternating weaving. By Theorem 4.2.3, the weaving σ is induced by an ERS ρ of a graph G. By Lemma 4.1.2,

we can assume without loss of generality that each edge is 0-twisted, 1^+-twisted, or 1^--twisted in ρ . If the ERS ρ is not uni-direction-twisted, then there must be edges $e, e' \in E_G$, such that e is 1^+-twisted and e' is 1^--twisted in ρ.

Let x and x' be the crossings in σ that are induced by the edges e and e', respectively. Since the σ-graph is connected, there is a sequence of strand segments (corresponding to a path in the σ-graph):

$$s_1, s_2, \ldots, s_q$$

such that

- strand segment s_1 starts from crossing x;
- strand segment s_q ends at crossing x';
- for $j = 1, \ldots, q-1$, strand segment s_j and strand segment s_{j+1} share a common crossing.

Since x is induced by the 1^+-twisted edge e and x' is induced by the 1^--twisted edge e', there must be a strand segment s_j in this sequence such that one crossing of s_j is induced by a 1^+-twisted edge and the other crossing of s_j is induced by a 1^--twisted edge. Now by an analysis similar to that in the proof of Theorem 7.1.1 (see Figure 7.2, where we showed that if both crossings of a strand segment are induced by two edges of the same twist type, then the strand segment proceeds alternatingly), we can verify that the strand segment s_j either goes over at both its crossings or goes under at both its crossings. This contradicts the premise that the weaving σ is alternating. □

Combining Theorem 7.1.1 and Theorem 4.2.3, we also have the following result.

Theorem 7.1.3 *Any normal weaving $\sigma : \mathcal{C} \to S_i$ can be converted into an alternating weaving by appropriately changing some edge-crossing types.*

PROOF. By Theorem 4.2.3, the weaving σ is induced by an ERS ρ of a graph G. Let ρ' be the ERS of graph G obtained from ρ by changing the twist type of each k^--twisted edge, where $k > 0$, to k^+-twisted. Then the ERS ρ' is uni-direction-twisted. By Theorem 7.1.1, the weaving σ' induced by ρ' is alternating. It is easy to see that the σ-imbedding and the σ'-imbedding are identical on the surface S_i. (More precisely, there is an autohomeomorphism on the surface S_i that maps the image of the σ-imbedding $\iota_\sigma : G_\sigma \to S_i$ to the image of the σ'-imbedding $\iota_{\sigma'} : G_{\sigma'} \to S_i$.) Therefore, the normal weaving σ can be converted into the alternating weaving σ' by properly changing some crossing types. □

Definition [1] A weaving is a ***twill weaving*** if it satisfies the following *Strand* and *Offset* conditions:(1) *Strand condition*: Every strand must traverse 2-up and 2-down consecutive crossings, alternatively. (2) *Offset condition*: For any strand segment s whose two ends are of the same type, the other ends of the two adjacent segments, which are s are on the boundary of the same gap, must have different types.

A weaving is ***twillable weaving*** if it can be converted into a twill by properly changing the cross types in σ.

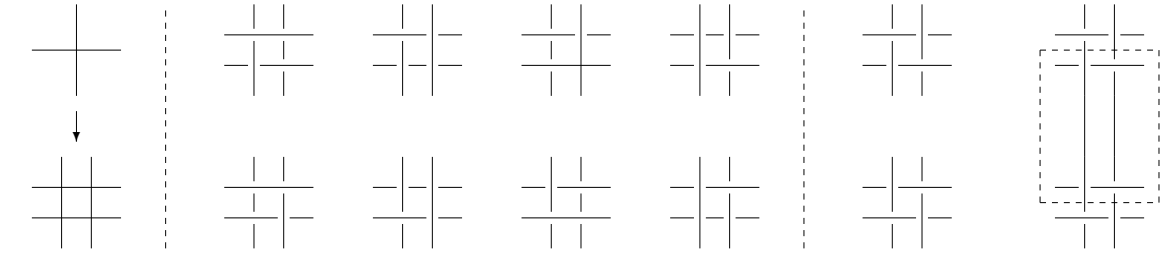
7.2 Child-Twillable Weaving

The ***child weaving*** of a weaving σ is the weaving that is obtained from σ by doubling all strands in σ.

Definition A weaving σ is ***child-twillable*** if its child weaving is a twillable weaving.

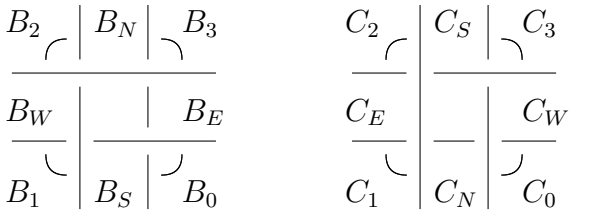
7.2.1 Identification of child-twillable weavings

We are most interested in identifying child-twillable weavings. Let σ_1 be a weaving and let σ_2 be the child of σ_1.



(a) "+" to "#" (b) feasible (c) non-feasible (d) non-twill

Figure 7.3: "#" structures after doubling strands



(a) B-# structure (b) C-# structure

Figure 7.4: Two basic feasible "#" structures

Each cross "+" in the weaving σ_1 is replaced by a "#" structure in the weaving σ_2 (see Figure 7.3(a)). In order to make σ_2 satisfy the Offset Condition of a twill, the four crosses of each # must be of one of the 10 configurations as given in Figures 7.3(c), then the two consecutive # structures in σ_2, which correspond to two consecutive crosses in σ_1, must be connected as shown in Figure 7.3(d) in order for each strand to satisfy the Strand Condition of a twill. However, this connected twill make the configuration between the two # structures violate the Offset Condition of a twill (as shown in the rectangular area in Figure 7.3(d)).

Therefore, the crosses of each # must be of one of the eight configurations as given in Figure 7.3(b). Each of these eight configurations can be obtained by properly rotating one of the two basic # structures, which will be names B-# and C-#, respectively, as given in Figure 7.4. Because of the importance of these two structures, we introduce some terminologies that will be used later. The B-# structure has four ***gates***: ***east gate*** B_E, ***south gate*** B_S, ***west gate*** B_W, and ***north gate*** B_N, ordered in clockwise order. Each gate of B-# consists of two parallel segments. We also name four ***corners*** B_0, B_1, B_2 and B_3 between the gates of B-#, where B_0 is between B_E and B_S, B_1 is between B_S and B_W, B_2 is between B_W and B_N, and B_3 is between B_N and B_E. For each corner we define an order of the two gates in clockwise order so that we enter the corner from the first gate and leave the corner from the second gate. For example, we enter corner B_0 from gate B_E and leaves it from gate B_S, and so on. See Figure 7.4 for detailed illustration of these notations. Similar terminologies are also given for the C-# structure, also shown in Figure 7.4. The configuration of a # structures is uniquely determined by its type (i.e., B-# or C-#) and a labeling of its gates (note that once a gate is labeled, the labellings of all other gates and all corner are also uniquely determined).

Two # structures are ***adjacent*** if they are directly connected via their gates. In order to make σ_2 twillable, the configurations of two adjacent # structures σ_1 and σ_2 must be consistent, i.e. the connection between σ_1 and σ_2 must keep the local structure to satisfy the Strand and Offset Conditions of a twill. For this, we have the following lemma.

Lemma 7.2.1 *The weaving σ_2 is twillable if and only if we can assign configurations to the # structures in σ_2 so that the configurations of any two adjacent # structures ξ_1 and ξ_2 satisfy the following conditions: (1) ξ_1 and ξ_2 must have different types (i.e. one is B-# and the other is C-#); and (2) ξ_1 and ξ_2 are connected via the gates, such that B_W, B_N, B_E, B_S are connected with C_E, C_S, C_W, C_N respectively.*

PROOF. Suppose that σ_2 is twillable. Then we can assign cross-types in σ_2 to make it a twill weaving. This gives a configuration to each # structure in σ_2. Let ξ_1 and ξ_2 be two adjacent # structures in σ_2. If ξ_1 is a B-# whose B_E is connecting to ξ_2, then it is easy to verify that ξ_2 must be a C-# whose C_W is connecting to ξ_1 (i.e., ξ_1 and ξ_2 must be connected via their west gates). We can similarly check all other cases and verify that in all cases, the configurations of ξ_1 and ξ_2 satisfy conditions (1) and (2) stated in the lemma.

For the other direction, suppose that we can assign configurations to the # structures in σ_2 so that the configurations of any two adjacent # structures satisfy conditions (1) and (2). It is again easy to verify, by checking the local structure in each # structures and the local structure between adjacent # structures, that this configuration assignment gives cross-types for σ_2 that make σ_2 satisfy the Strand and Offset Conditions of a twill weaving. □

Thus, if the configurations of two adjacent # structures in σ_2 satisfy conditions

(1) and (2) in Lemma 7.2.1, we say that the configurations are ***consistent***.

There is another way to interpret Lemma 7.2.1, in terms of walks in a weaving.

Definition A ***weave-walk*** in a weaving σ is a sequence of oriented segments $(s_0, s_1, \ldots, s_k)$ in σ such that for all i, $0 \leq i \leq k-1$, the tail of s_i and the head of s_{i+1} are at the same cross. The walk is ***closed*** if the tail of s_k and the head of s_1 are also at the same cross.

Note that by definition, two oriented segments s_1 and s_{i+1} consecutive on a weave-walk may belong to two different strands(i.e., the weave-walk "jumps" from the segment s_i of a strand to a segment s_{i+1} of another strand at a cross).

Definition The ***corner cost*** of a weave-walk w at a cross x in a weaving σ is the number of corners, when we rotate around x in clockwise order, between the segment from which w enters the cross and the segment from which w leaves the cross. (see Figure 7.5)

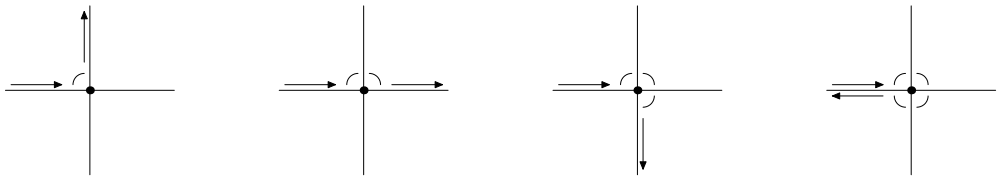


Figure 7.5: Corner costs at a cross

Note that a cross x may appear more than once in the weave-walk w. In this case, each appearance of x on w is treated as a different cross and its corner cost is computed separately based on the entry segment and the exit segment for the cross, by the definition.

Definition The ***corner cost*** of a weave-walk w in a weaving σ is the sum of corner costs of w on the crosses on w. The ***segment-length*** of w is the number of segments traversed by w.

Recall that we leave the corner B_3 of a B-# ξ_1 from its east gate and enter the corner C_0 of a C-# ξ_2 from its west gate. Therefore, if ξ_1 and ξ_2 are adjacent, have consistent configurations, and are connected via their east-west gates, then we can regard this as a traverse from the corner B_3 of ξ_1 to the corner C_0 of ξ_2. This observation gives a precedence relation for the corners of the B-# and C-# structures. We examine all such corner relations and represents the ***corner precedence graph*** (abbr. CPG) D_0, as given in Figure 7.6.

A directed edge from B_i to $C_{(i+1) \ mod \ 4}$ in the CPG D_0, $0 \leq i \leq 3$, represents the corner move from the corner B_i of a B-# to the corner $C_{(i+1) \ mod \ 4}$ of an adjacent C-# (recall that for each corner, we have an entry gate and an exit gate). A directed edge from C_i to $B_{(i+1) \ mod \ 4}$ is interpreted similarly. An edge in D_0 is a ***transition edge*** if it is between a B vertex and a C vertex. Transition edges are given by thin vectors in Figure 7.6. There are also edges between two B vertices and between two C vertices in D_0, called ***corner-transition edge*** and given by thick vectors in Figure 7.6, which represent the corner moves in the same # structures. For example, the edge from B_0 to B_1 represents the corner move from the corner B_0 to the corner

B_1 in the same B-# (i.e. we leaves the corner B_0 from the south gate B_S and enter the corner B_1 also from the south gate B_S in the same B-#).

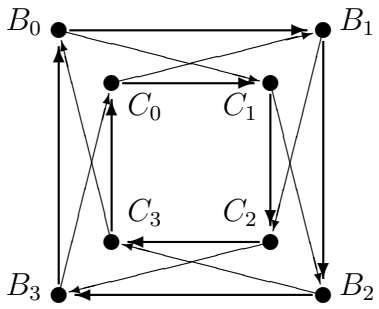


Figure 7.6: The corner precedence graph (CPG) D_0

A ***edge-walk*** in CPG D_0 is an ordered sequence $w = (e_1, e_2, \ldots, e_s)$ of directed edges such that for all $1 \leq i \leq s-1$, the tail of the directed edge e_i and the head of the directed edge e_{i+1} are at the same vertex. The edge-walk w is *closed* if the tail of e_s and the head of e_1 are also at the same vertex. It is easy to verify the following property.

Lemma 7.2.2 *An edge-walk w in the weaving σ_1 in CPG D_0 is closed if and only if the number of edges traversed by w is a multiple of 4 and if the number of transition edges traversed by w is even.*

Each weave-walk w_1 in the weaving σ_1 induces a #-walk w_2 in the weaving σ_2, which consists of a sequence of adjacent # structures corresponding to the crosses in w_1. The #-walk w_2 in σ_2 is *closed* if the weave-walk w_1 in σ_1 is closed.

Now we are ready for our main theorem for this section.

Theorem 7.2.3 *A weaving σ_1 is child-twillable if and only if for each closed weave-walk w_1 in σ_1, the corner-costs of w_1 is divisible by 4 and the segment-length of w_1 is divisible by 2.*

PROOF. Let σ_2 be the child weaving of the weaving σ_1. The child-twillability of σ_1 is equivalent to the twillability of σ_2.

To prove the first direction, suppose that σ_2 is twillable. Thus, without loss of generality, we can assume that σ_2 itself is a twill.

Let w_1 be any closed weave-walk in σ_1, starting from a cross x. There is a corresponding closed #-walk w_2 in σ_2, starting from a corresponding #-structure ξ. Since σ_2 is a twill, ξ must be of one of the eight configurations given in Figure 7.3(2). Thus, the type, gates, and corners of ξ are all uniquely determined. By Lemma 7.2.1, the configuration of any # structure adjacent to ξ is also uniquely determined. Inductively, if we traverse the #-walk w_2, starting from ξ, then the configurations of all # structures on w_2 in σ_2 can be recursively and uniquely determined.

The closed #-walk w_2 can be traced by the corners traversed by w_2. which has a corresponding edge-walk w' in the CPG D_0. In particular, each vertex traversed by w' corresponds to a corner of a # structure in w_2 and each edge traversed by w' corresponds to a move from a corner to another corner in w_2. Since w_2 is a closed #-walk in the twill σ_2, w_2 must finally come back to the starting corner of the # structure ξ. As a result, the corresponding edge-walk w' in the CPG D_0 must be closed (i. e., it starts and ends at the same vertex in D_0). By Lemma 7.2.2, the number of edges traversed by w' is a multiple of 4 and the number of transition edges traversed by w' is even.

On the other hand, the number of edges traversed by the closed edge-walk w'

in D_0 corresponding to the number of corners traversed by the closed #-walk w_2 in σ_2, and the number of transition edges traversed by w' is equal to the number of transitions between different # structures in the traversing of the #-walk w_2. Since there is a one-to-one correspondence between the corners in w_2 and the corners in w_1, and there is a one-to-one correspondence between the transitions between different # structures in w_2, and the segments in w_1, the number of edges traversed by the closed edge-walk w' in D_0 is equal to the corner cost of the closed weave-walk w_1, and the number of transition edges traversed by w' is equal to the segment-length of w_1. This proves that the corner-length of the closed weave-walk w_1 in σ_1 is divisible by 4, and that the segment-length of w_1 is divisible by 2. Since w_1 is an arbitrarily picked closed weave-walk in the weaving σ_1, this completes the proof of the first direction.

To prove the other direction, suppose that for every closed weave-walk w_1 in σ_1, the corner cost of w_1 is divisible by 4 and the segment-length of w_1 is divisible by 2. We show that the child weaving σ_2 is twillable.

Pick any # structure ξ in σ_2 that corresponds to a cross in σ_1 and assign ξ an arbitrary configuration $c(\xi)$. We can extend this configuration assignment to cover the entire weaving σ_2 under the assumption that any two crosses in σ_1 are connected by a weave-walk: for any two adjacent # structures ξ_1 and ξ_2 where the configuration $c(\xi_1)$ of ξ_1 has been assigned, we can uniquely determine the configuration $c(\xi_2)$ of ξ_2. Let us call this process of configuration assignment the A_1***-process***.

We must verify that the A_1-process is consistent. For this, suppose inductively that so far the configurations assigned by the A_1-process are consistent. Let ξ' be a # structure in σ_2 such that ξ' has not been assigned a configuration, and that ξ' is adjacent to two # structures ξ_s and ξ'_t that has been assigned configuration $c(\xi_s)$ and $c(\xi'_t)$, respectively. We must show that there is a unique configuration for ξ' that

is consistent with both $c(\xi_s)$ and $c(\xi'_t)$.

For this, let $w_s = (\xi_0, \xi_1, \ldots, \xi_s)$ be the #-walk from ξ to ξ_s, where $\xi_0 = \xi$, and for all $0 \leq i \leq s-1$, the configuration $c(\xi_{i+1})$ of ξ_{i+1} is determined by the configuration $c(\xi_i)$ of ξ_i. Also, let $w_t = (\xi'_t, \xi'_{t-1}, \ldots, \xi'_0)$ be the #-walk from ξ'_t to ξ, where $\xi'_0 = \xi$, and for all $1 \leq i \leq t$, the configuration $c(\xi'_i)$ of ξ'_i is determined by the configuration $c(\xi'_{i-1})$ of ξ'_{i-1}. Then

$$w_2 = (\xi_0, \xi_1, \ldots, \xi_s, \xi', \xi'_t, \xi'_{t-1}, \ldots, \xi'_1, \xi'_0)$$

is a closed #-walk in σ_2. This #-walk w_2 corresponds to a closed weave-walk w_1 in σ_1. By the assumed conditions, the closed weave-walk w_1 has its corner cost divisible by 4 and segment length divisible by 2. By the relationship between w_1 and w_2, we derive that the number of corners traversed by the #-walk w_2 is divisible by 4 and w_2 contains an even number of transitions between different # structures. Now if we let w' be the edge-walk in the CPG D_0 that corresponds to the #-walk w_2 in σ_2, then the number of edges traversed by the edge-walk w' in D_0 is divisible by 4 and the number of transition edges traversed by w' is divisible by 2. By Lemma 7.2.1, w' is a closed edge-walk.

Therefore, if we apply the following process of configuration assignments on the # structures on w_2 (called A_2***-process***): start with the configuration $c'(\xi) = c(\xi)$ of $\xi = \xi_0$, traverse the #-walk w_2 and assign an unique configuration $c'(\xi')$ to each encountered # structure based on the configuration of the # structure before, then we will get configuration assignments for all # structures on w_2 such that the configurations of any two adjacent # structures on w_2 are consistent. In particular, the configuration $c'(\xi'_1)$ of ξ'_1 is consistent with the configuration $c'(\xi) = c(\xi)$ of ξ, because the corresponding edge-walk w' in D_0 is closed. Note that for the # structures

$\xi_0, \xi_1, \ldots, \xi_s$ in the #-walk w_s, the A_1-process and the A_2 process must assign the same configurations: they both start with the same configuration $c(\xi)$ and assign a configuration to the next # structure recursively, in the same order. In consequence, if the A_1-process assigns the configuration of ξ' based on $c(\xi_s)$, then ξ' will be assigned the configuration $c'(\xi')$.

Consider two connected # structures ξ_1 and ξ_2. By the definition of configuration consistence, if we start with a configuration $c_1(\xi_1)$ of ξ_1, then the configuration $c_2(\xi_2)$ of ξ_2 is uniquely determined. This implies that if we start with the configuration $c_2(\xi_2)$ of ξ_2, then the configuration of ξ_1 is also uniquely determined: it must be $c_1(\xi_1)$. Therefore, if we traverse the "reverse" #-walk of w_2:

$$w_2' = (\xi_0', \xi_1', \ldots, \xi_t', \xi', \xi_s, \xi_{s-1}, \ldots, \xi_1, \xi_0)$$

starting with the configuration $c'(\xi) = c(\xi)$ of $\xi = \xi_0$ and assigning the configuration of the next encountered # structure based on that of the previous # structure, we should still get configuration $c'(\xi_i')$ for ξ_i', for $0 \leq i \leq t$, and get configuration $c'(\xi')$ for ξ'. However, this is exactly the order the A_1-process assigns the configurations for $\xi_0', \xi_1', \ldots, \xi_t'$. Therefore, we must have $c'(\xi_i') = c(\xi_i')$ for $0 \leq i \leq t$. Moreover, the configuration $c'(\xi')$ of ξ' is determined by the configuration $c(\xi_i')$ of ξ_i'. In consequence, if the A_1-process assigns the configuration of ξ' based on $c(\xi_t')$, then ξ' will be assigned the same configuration $c'(\xi')$.

This completes the proof that there is an unique configuration $c'(\xi')$ for the # structure ξ' that is consistent with both the configurations $c(\xi_s)$ and $c(\xi_t')$.

Now we are ready to complete the proof of the theorem. Starting from any # structure ξ and giving it an arbitrary configuration, the A_1-process recursively assigned an unique configuration to each # structure that is adjacent to a # structure

that has been assigned a configuration. The above analysis shows that the configuration assignment by the A_1-process satisfies the condition that the configurations of any two adjacent # structures are consistent. By Lemma 7.2.1, the weaving σ_2 is twillable. □

The second part of the proof of Theorem 7.2.3 actually shows that whether σ_2 is twillable or not is dependent of the selection of the first # structure ξ and of the configuration assigned to ξ. Therefore, we immediately present a general algorithm in Figure 7.7 that tests if a given weaving σ_1 is child-twillable.

Placement(σ_1)
INPUT: a weaving σ_1 on a surface;
OUTPUT: "yes" if σ_1 is child-twillable, "no" otherwise;
1. pick any cross x in σ_1 and assign it an arbitrary configuration;
2. **if** Config(x) = false **then** return "no";
3. return "yes";

Config(x)
1. **for** each cross x' adjacent to x **do**
2.1 **case 1.** the configuration of x and x' are not consistent
return false;
2.2 **case 2.** x' has not been assigned a configuration
assign the consistent configuration to x';
if Config(x') = false **then** return false;
3. return true;

Figure 7.7: The algorithm *Placement*

The correctness of the algorithm *placement* is ensured by the same argument as our analysis for the twillability of σ_2 given in the proof of Theorem 7.2.3. Note that the algorithm returns at step 3 only if all crosses in σ_1 are assigned configurations

and no inconsistency is discovered. If the algorithm stops in the recursive execution of the subroutine Config(x) initialted in step 2, it will claim correctly that σ_1 is not child-twillable.

Theorem 7.2.3 also provides some convenient methods to test the (non) child-twillability of a weaving σ.

Corollary 7.2.4 *If a weaving σ has a gap whose size is not divisible by 4, then σ is not child-twillable.*

PROOF. The boundary walk of a gap forms a weave-walk in σ whose corner cost is equal to the size of the gap. Thus, if the gap size is not divisible 4, then, by Theorem 7.2.3, the weaving σ is not child-twillable. □

A strand c in a weaving σ is ***self-intersecting*** if there is a cross x in σ such that all four oriented segment meeting at x from the strand c.

Corollary 7.2.5 *If a weaving σ has a self-intersecting strand, then $\sigma$is not child-twillable.*

PROOF. Let c be a self-intersecting strand in the weaving σ such that two segments s_1 and s_2 of c meet at a cross x and make a corner of x. Starting from the cross x and following the segment s_1, traverse the strand c until we come back to the cross x from the segment s_2. This makes a closed weave-walk w in σ. The closed weave-walk w has a corner cost of 2 at all crosses on w, except at the cross x, at which the corner cost is either 1 or 3. Therefore, the corner cost of w is an odd number. By Theorem 7.2.3, the weaving σ is not child-twillable. □

Corollary 7.2.6 *Every weaving on the sphere is not child-twillable.*

PROOF. Let σ be a weaving on a sphere. Let G_σ be a graph obtained from σ whose vertices correspond to the crosses in σ and whose edges correspond to the segments in σ. Then G_σ is a graph imbedded in the sphere. Let n, m and f be the number of vertices, the number of edges, and the number of faces of the graph G_σ imbedded on the sphere. Since every vertex in G_σ has degree 4, we have $4n = 2m$, which gives $n = m/2$.

By Euler's formula, we have $n - m + f = 2$. Replace n by $m/2$, and rearrange the formula, we get $f = m/2 + 2$. On the other hand, if we add all face sizes together, we could get $2m$ because each edge is used exactly twice in face boundaries. Therefore, the "average face size" is equal to $2m/f = (2m)/(m/2 + 2) < 4$. Therefore, there is at least one face whose size is less than 4, thus not divisible by 4. Since each face in G_σ corresponds to a gap in σ, this shown that the weaving σ has a gap whose size is not divisible by 4. By Corollary 7.2.4, the weaving σ is not child-twillable. □

We close this section by another interesting and useful property of the child-twillable weavings. Let σ_1 be a child-twillable weaving and let σ_2 be the child of σ_1 and σ_2 is a twill. Recall from Lemma 7.2.1 that two adjacent # structures in σ_2 must be connected via the consistent gates. Therefore, if we traverse a strand c in σ_2, the strand c must either only go through the east and west gates of the # structures, or only go through the south and the north gates of the # structures in σ_2. Therefore, the strands in σ_2 can be partitioned into two disjoint groups: one group contains those that only go through east and west gates of the # structures and the other group contains those that only go through south and north gates of the # structures. Note that no two strands in the same group can make a cross. In other words, every

cross in σ_2 is made by two strands that belong to different groups. Therefore, the strands in σ_2 can be classified as the "weft threads" and the "warp threads" such that each cross is make by one weft thread and one warp thread. This nicely generalizes the corresponding concepts in planar twill structures. In particular, we can paint the two groups using two different colors that make the diagonal patterns in a twill look nicer. We summarize this discussion in the following theorem.

Theorem 7.2.7 *Let σ_1 be a child-twillable weaving and let σ_2 be the child of σ_1 and σ_2 is a twill. Then the strands in σ_2 are can be partitioned into weft strands and warp strands such that every cross in σ_2 is made by a weft strand and a warp strand.*

7.2.2 Descendants of child-twillable weavings

We have shown how to construct a weaving structure based on the extended rotation systems. Let M_1 be a mesh on a surface S. Applying the well-known Catmull-Clark subdivision algorithm[10] or Doo-Sabin subdivision algorithm[16] on M_1, we obtained a refined mesh M_2. Let σ_1 be the weaving induced from the mesh M_1, and let σ_2 be the weaving induced from the mesh M_2. We have shown that the weaving σ_2 is the child weaving of σ_1. Suppose that the weaving σ_2 is twillable, the weaving σ_1 is child-twillable. It will be interesting to know whether the child weaving σ_2 is also child-twillable. In particular, if all the weavings obtained by recursively doubling the strands in a resulting weaving are child-twillable, then we can repeatedly apply Catmull-Clark subdivision algorithm or Doo-Sabin subdivision algorithm on the meshes and get finer and finer twill weaving structures on the surface S.

We start with some interesting structures of a child-twillable weaving. Let w be a weaving-walk in a weaving σ. We say that a cross x on w is of ***even-cost*** (resp. ***odd-cost***) if the corner cost of w at x is either 0 or 2 (resp. either 1 or 3).

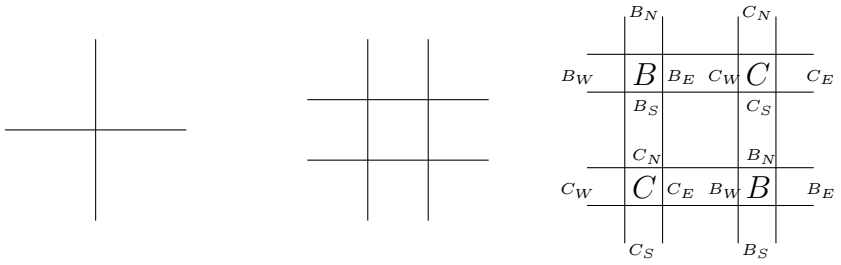
Lemma 7.2.8 *Let σ be a child-twillable weaving on a surface S, and let w be a closed weave-walk in σ. Then w passes through an even number of even-cost crosses, as well as an even number of odd-cost crosses.*

PROOF. By Theorem 7.2.3, the segment-length of w is divisible by 2. Therefore, the closed weave-walk w passes through an even number of crosses. If the number of even-cost crosses on w is odd, then the number of odd-cost crosses on w is also odd. Since the total corner costs on any set of even-cost crosses is an even number, and the total corner costs on an odd number of odd-cost crosses is an odd number, this would imply that the corner costs of the closed weave-walk w is an odd number, contradicting the assumption that the weaving σ is a child-twillable by Theorem 7.2.3. This proves that w must pass through an even number of even cost crosses. Since the total number of crosses on w is even, we also derive that w passes through an even number of odd-cost crosses. □

Now we are ready for our main theorem in this subsection.

Theorem 7.2.9 *Let σ_1 be a child-twillable weaving and let σ_2 be a child of σ_1. Then σ_2 is child-twillable.*

PROOF. According to Lemma 7.2.1, we only need to show that the child of σ_2 has a consistent # configuration to complete the proof. Let σ_3 be the child of σ_2. For each cross in σ_1, there is a corresponding # structure in σ_2, and four corresponding # structures in σ_3, as shown in Figure 7.8. We name the four # structures as an ***unit***, where an unit connects with other units via eight gates. In the following, we show there exists a configuration for σ_3, such that the # structures inside each unit are consistent and gates between any two adjacent units are consistent.



(a) a cross in σ_1 (b) a # structure in σ_2 (c) four # structures in σ_3

Figure 7.8: Descendants of child-twillable structures

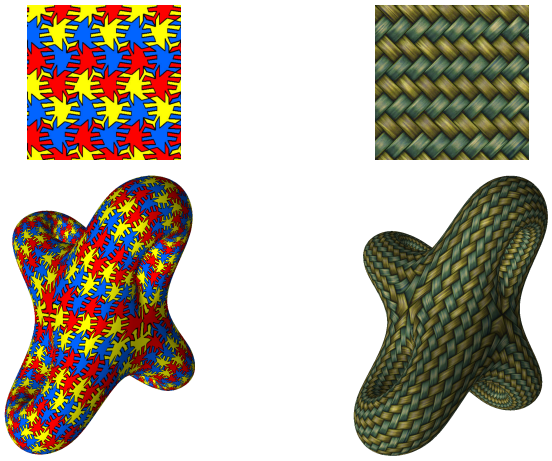
We configure an unit as following: we assign an arbitrary configuration to the top-left # structure in the unit, then we recursively assign the consistent configurations to the other three # structures. It is easy to verify that in all cases the four # structures can be consistent inside an unit. Without loss of generality, we assume the top-left # structure in an unit is assigned as B-#, and the top gate of B-# is B_N, as shown in Figure 7.8(c), where the # structures in an unit are consistent. Moreover, the top gates B_N, C_N of the unit are consistent with the bottom gates C_S, B_S, if we duplicate and transit the unit vertically. Similarly, the left gates B_W, C_W are consistent with the right gates C_E, B_E if we duplicate and transit the unit horizontally. We show that if we duplicate and transit a consistently configured unit through east-west gates and south-north gates, we can cover σ_3 without a conflict. In Theorem 7.2.7, we already discussed that the strands in σ_2 can be divided into two groups, weft strands and warp strands, where each cross is made with one weft strand and one warp strand. Since σ_3 is obtained from σ_2 by doubling the strands, we can also separate the strands in σ_3 into two groups and each cross is made by a strand from each group. It is also safe to say that, in a consistently configured unit, the strands go through east-west gates are from one group and the strands go through south-north gates are from the other group. Therefore, an unit transits

through east-west gates would never meet south-north gates in σ_3; on the other hand, an unit transits through south-north gates would never meets east-west gates. The weaving σ_3 has a consistent configuration. □

The proof of Theorem 7.2.9 actually presents interesting properties of the grand-child σ_3 of a child-twillable weaving σ_1: the weaving σ_3 is constructed with the transited copies of a consistently configured unit; the left and right sides of the unit are consistently matched; the top and bottom sides of the unit are consistently matched. In the following, we generalize the property to a new type of imbedded graphs, and realize its applications in some interesting computer graphics problems.

8. QUAD-PATTERN COVERABLE MESHES*

As a popular area in computer graphics, texture mapping[9] can create complicated and beautiful images on surfaces. Especially, the texture mapping with repetitive patterns requires limited memory costs. In this chapter[26], we introduce a quadrangular mesh which can be seamlessly mapped with one single quadrangular pattern, where the pattern can be seamlessly mapped on a torus, as shown in Figure 8.1.



(a) Escher-like pattern (b) Twill-weave pattern

Figure 8.1: The quadrangular meshes covered with a periodic quad-pattern[26]

A ***quad-mesh*** is a mesh in which all faces are 4-sided, and in an n***-regular mesh***, every vertex has valence n. Let P be a rectangle pattern (i.e., an image that fills a rectangle). We name its four sides East (E-), South (S-), West (W-), and

*reprinted with permission from "Pattern mapping with quad-pattern-coverable quad-meshes" by Shiyu Hu, Qing Xing, Ergun Akleman, Jianer Chen, Jonathan L. Gross, 2012. Computers & Graphics 36(5): 455-465, Copyright[2012] by Elsevier.

North (N-), so that a clockwise traversal of its boundary encounters the sides in that order. The pattern P is *periodic* if its mapping to a torus is seamless.

Definition A quad-mesh M is ***quad-pattern coverable*** (abbr. ***QPC***) if its faces can be covered by a periodic rectangle pattern P, such that for any two adjacent faces of M, the common edge between them is either the E-side of one of the two faces and the W-side of the other, or else the S-side of one of the two faces and the N-side of the other.

Note that the quad-pattern coverability is a property of a QPC mesh, which is independent with the chosen patterns.

8.1 Identification of QPC Meshes

Recall that for a mesh M on a surface S, the ***dual mesh*** M' is a graph imbedded on the same surface S, such that

- there is a bijective correspondence between the faces of M and the vertices of M' and a bijective correspondence between the edges of M and the edges of M';
- if faces f_1 and f_2 have edge e in common, then the dual edge e' joins the vertices f_1' and f_2' of M' that are dual to f_1 and f_2, respectively.

The four oriented edges incident to each vertex f' of M' can be labeled (E-), (S-), (W-), and (N-), in a cyclic ordering consistent with the rotation at f'. A collection of such labellings, one for each vertex of M', is called an *oriented-edge labeling* of M'. The four labels around each vertex f' induce labels of the face f to which f' is dual. Based on this observation, we introduce the following definition.

Definition Let M' be a 4-regular mesh imbedded on a surface S. An oriented-edge labeling of M' is ***consistent*** if for every oriented edge labeled (E-), (S-), (W-), or (N-), the oppositely oriented edge is labeled (W-), (N-), (E-), or (S-), respectively.

We immediately infer the following theorem.

Theorem 8.1.1 *A quad-mesh M is quad-pattern coverable if and only if its dual mesh M' can be labeled consistently.*

Recall the proof of Theorem 7.2.9, it is easy to realize that if the σ-imbedding ι_σ of a child-twillable weaving is a cellular imbedding, ι_σ is the dual of a QPC mesh. We view each unit as a period pattern, and view each cross as the dual vertex of a quadrangular face. By replacing each cross with a consistently configured unit, we can construct a seamlessly twill weaving. The following theorem gives more clear reference.

Theorem 8.1.2 *Let M be a 4-regular mesh imbedded on a surface S. Then mesh M can be labeled consistently if and only if for every closed walk in M, the corner-cost of that walk plus twice its length is divisible by* 4.

PROOF. Let β be a closed walk in the mesh M with corner-cost q and length k such that $q + 2k$ is divisible by 4. Quite simply, this means

if k is even, then $q \equiv 0 \mod 4$, and

if k is odd, then $q \equiv 2 \mod 4$.

To help to analyze the corner-cost of a walk β in the mesh M, we create a similar corner precedence graph D_1, as shown in Figure 8.2.

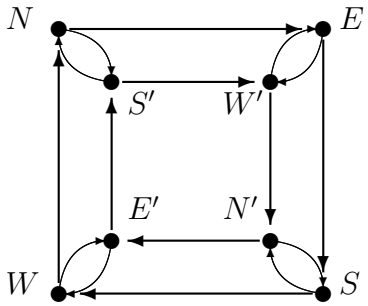


Figure 8.2: The corner precedence graph (CPG) D_1

($\Rightarrow$) Suppose first that mesh M has a consistent labeling L, and that we have a walk

$$\beta = \{e_1, e_2, \ldots, e_k\}$$

in M, where $e_i = \langle v_i, v_{i+1} \rangle$ is an oriented edge for each i (with $v_{k+1} = v_1$). We now construct a longer walk β' in the corner-graph D, which has some "corner-steps" between consecutive β-steps that we denote by e'_i.

We can assume, without loss of generality, that the oriented edge e_1 is the E-edge of vertex v_1. Then we take vertex E as the starting vertex of the walk β' in D. Since the labeling L is consistent, the oriented edge $\overline{e}_1 = \langle v_2, v_1 \rangle$ must be the W-edge of vertex v_2. We take the directed edge e'_1 of walk β' in corner-graph D to be the transition-edge from vertex E to vertex W', which signifies that after walk β traverses the E-edge $e_1 = \langle v_1, v_2 \rangle$ out of v_1, it next takes the W-edge $\overline{e}_1 = \langle v_2, v_1 \rangle$ into v_2. Now suppose that the walk β has corner-cost h at the vertex v_2. Then we let walk β' traverse h directed corner-edges on the inner rectangle in the corner-graph, after which we observe that the walk β' must be at a vertex of D whose name matches the label of the oriented edge $e_2 = \langle v_2, v_2 \rangle$ of mesh M. In general, each maximal subsequence of consecutive corner-edges in β' corresponds to the corner-cost of the walk β as it passes through a vertex in M.

If the walk β is closed, then it returns to the origin of the oriented edge e_1. As a consequence, the directed walk β' in D ends up either at the vertex E or at the vertex E' (recall that we assumed that e_1 is the E-edge of v_1).

If the length k of walk β is even, then the walk β' contains an even number of transition-edges, so it ends up at the vertex E, and it is closed. The structure of the corner-graph D implies that the number of corner-edges in a closed directed walk is divisible by 4. Thus, $q + 2k$ is divisible by 4. Alternatively, if the length k is odd, then the walk β' contains an odd number of transition-edges, and thus it ends up at the vertex E'. In this case, the structure of the corner-graph D implies that the number of corner-edges in a closed directed walk is congruent to 2 mod 4. Thus, once again, the number $q + 2k$ is divisible by 4. This completes the proof for this direction.

($\Leftarrow$) For the other direction, suppose that every closed walk in M satisfies the condition given in the theorem. We show how to construct a consistent labeling on the oriented edges of the mesh M.

Pick a root vertex v of mesh M, and assign the labels E-, S-, W-, and N- to the oriented edges originating at v, in a manner consistent with the orientation of surface S. Proceeding inductively, suppose that a labeling at the vertex w_1 has been assigned and that the labeling for the neighboring vertex w_2 has not been assigned. Then assign to the edges originating at vertex w_2 the unique four labels that are consistent with those at w_1. Continue until all oriented edges of mesh M are labeled. We must verify that this labeling process L is consistent. For this, it is sufficient to show that the labeling at a vertex w does not depend on the choice of a labeled neighbor vertex.

Suppose, as an inductive hypothesis, that so far, the labellings for a set of vertices

are consistent. Let w be an as yet unlabeled vertex, with neighbors w_1 and w_2 of w that have been assigned the labellings $l(w_1)$ and $l(w_2)$, respectively. We will show that the labeling $l(w_1)$ and the labeling $l(w_2)$ would induce the same labeling for the vertex w.

For this, we let $\beta_1 = \{v_0 = v, v_1, \ldots, v_r = w\}$ be a walk of corner-cost p from the root vertex v to w_1 and $\beta_2 = \{u_s = w, u_{s-1}, \ldots, u_0 = v\}$ a walk of corner cost q, such that all vertices in β_1 and β_2 are already labeled. Then

$$\beta_3 = \{v_0 = v, v_1, \ldots, v_r = w = u_s, u_{s-1}, \ldots, u_0 = v\} \tag{8.1}$$

is a closed walk in M. By the premise for this direction of the theorem, if the length $r + s$ of β_3 is even, then the corner-cost $p + q$ of β_3 is divisible by 4; and if $r + s$ is odd, then $p + q$ is divisible by 2, but not by 4.

Case 1. We first suppose that the length $r+s$ of walk β_3 is even and that $p+q \equiv 0$ mod 4. Since $p + q \equiv 0 \mod 4$, it follows that the walk β_2^{-1} has corner cost $4 - p$ mod 4, and in turn, that the walk β_2 has corner cost $p \mod 4$. Since, furthermore, the lengths r and q are of the same parity, it follows that the induced walks β_1' and $(\beta_2^{-1})'$ terminate at the same vertex of the corner graph, and accordingly, that they induce the same labellings at w.

Case 2. Now suppose that the length $r + s$ of walk β_3 is odd and that $p + q \equiv 2$ mod 4. Since $r+s$ is odd, it follows that r and s are of different parity, which implies that one of the induced walks β_1' and $(\beta_2^{-1})'$ in the corner graph terminates on the inner 4-cycle and the other on the outer 4-cycle. Since $p + q \equiv 2 \mod 4$, it follows that the locations of these termination vertices are diagonally opposite. Thus, the these termination vertices have the same label, except that one of them is marked *prime*. It follows that the induced walks β_1' and $(\beta_2^{-1})'$ induce the same labellings at

w.

This completes the proof of the theorem. □

Theorem 8.1.2 immediately implies the following result.

Corollary 8.1.3 *A quad-mesh M is quad-pattern coverable if and only if for every closed walk β in the dual mesh M', the corner-cost of β plus twice its length is divisible by* 4.

Corollary 8.1.4 *If σ-imbedding ι_σ of a child-twillable weaving is cellular, ι_σ is the dual of a QPC mesh.*

With a little effort, we realize some properties of child-twillable weaivngs are preserved for QPC meshes.

Corollary 8.1.5 *There is a linear-time algorithm that tests whether a quad-mesh M is quad-pattern coverable, and when M is quad-pattern coverable, constructs a quad-pattern covering for the mesh.*

Lemma 8.1.6 *Let M be a 4-regular mesh imbedded on a surface, with consistent labellings. Then the size of each face is divisible by* 4.

Lemma 8.1.7 *Every mesh on a sphere is not a QPC mesh.*

Lemma 8.1.8 *If a mesh M is quad-pattern coverable, then the mesh M_c after Catmull-Clark subdivision on M is also quad-pattern coverable.*

We will not provide a complete proof for this lemma. Catmull-Clark subdivision turns each quad into 2×2. Therefore, applying Catmull-Clark subdivision is the same as using 2×2 version of a given quad-pattern. It is, therefore, the boundaries

will still match. The lemma is useful since it shows that any surface of positive genus can iteratively be covered by finer versions of a given texture.

Lemma 8.1.9 *The number irregular vertices, which are not 4-valences, of a QPC mesh is bounded by* $2(g-1)$ *on a surface with genus* $g > 1$.

PROOF. Since each face in a QPC mesh M is 4-sided, we have $4|F| = 2|E|$, where $|F|, |E|$ are the number of faces and edges in M. According to the Euler's formula, we have $|V| - |E|/2 = 2 - 2g$, where $|V|$ is the number of vertices in M. By Lemma 8.1.6, the valence of each vertex is a multiple of 4. Thus, the sum of all vertices' valences is $4|V| + R = 2|E|$, where R is the sum of extra degree of irregular vertices. Combining the equations $4|V| + R = 2|E|$ and $|V| - |E|/2 = 2 - 2g$, we have the sum of extra degree of irregular vertices is $8(g-1)$. The number of irregular vertices can be at most $2(g-1)$ when all of them are 8-valent. □

8.2 QPC-Preserving Valence Reduction

Unlike the general quadrilateral mesh editing operations [29, 41], we use dual meshes to reduce vertex valences in a QPC mesh, while keeping the QPC property. Let F be a $4k$-sided face in a 4-regular mesh M on a surface S, where $k \geq 2$ is a positive integer. Let

$$\beta_F = \{v_1, v_2, \ldots, v_{4h-1}, v_{4h}, v_{4h+1}, v_{4h+2}, \ldots, v_{4k-1}, v_{4k}\}$$

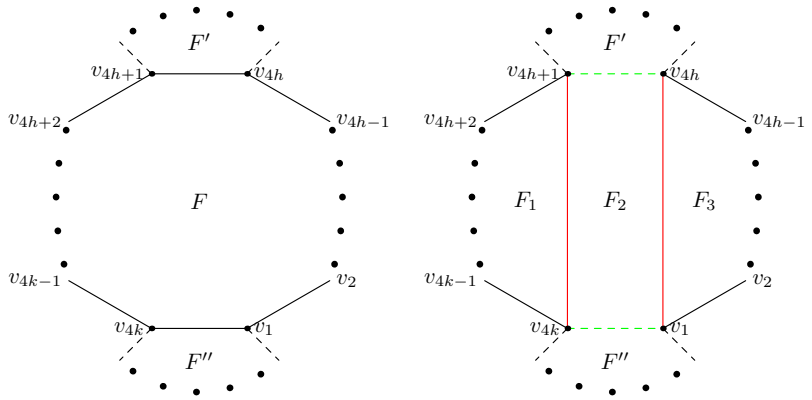
be the boundary walk of face F, in counterclockwise order, where $1 \leq h \leq k-1$. The (F, k, h)-*operation* on the mesh M, illustrated in Figure 8.3, is defined as follows:

- insert two non-crossing edges $[v_1, v_{4h}]$ and $[v_{4h+1}, v_{4k}]$ into the face F so as to split face F into three faces F_1, F_2, and F_3, where the "middle face" F_2 has

size 4; and

- delete the edges $[v_{4h}, v_{4h+1}]$ and $[v_{4k}, v_1]$ so as to merge the new middle face F_2 with two other faces F' and F''.

Note that if the mesh M is 4-regular, then the resulting mesh is also 4-regular.



(a) Before (b) After

Figure 8.3: Reducing face-size

The (F, k, h)-operation eliminates the face F of large size. However, merging the new middle face F_2 with faces F' and F'' may create another large face with $n' + n'' - 2 + 2$ faces, where n', n'' are the face-sizes of F' and F''. Nonetheless, if F is an "isolated" large face (i.e., if the faces adjacent to face F are all small), or if there is even a single appropriately located pair F' and F'' with small n', n'', then the (F, k, h)-operation effectively reduces the large-face problem. For example, suppose that the face F is 16-sided (i.e., $k = 4$) and that for $h = 2$, faces F' and F' are both 4-sided. Then the (F, k, h)-operation replaces F, F', and F'' by three 8-sided faces.

Lemma 8.2.1 *Let M be a 4-regular mesh imbedded on a surface S with consistent*

labelings, and let F be a $4k$-sided face of M, where $k \geq 2$. Then for any h with $1 \leq h \leq k-1$, the (F,k,h)-operation on M will result in a 4-regular mesh with consistent labellings.

PROOF. Let L_M be a consistent labeling for the mesh M. Consider the boundary walk (in counterclockwise order)

$$\beta_F = \{v_1, v_2, \ldots, v_{4h-1}, v_{4h}, v_{4h+1}, v_{4h+2}, \ldots, v_{4k-1}, v_{4k}\}$$

as in Figure 8.3(a). Suppose that the oriented edge $\langle v_1, v_{4k}\rangle$ is the E-edge for the vertex v_1 under the labeling L_M. (The other three cases have the same proof.) Then the oriented edge $\langle v_{4k}, v_1\rangle$ is the W-edge at vertex v_{4k}. From this we know that $\langle v_1, v_2\rangle$ is the S-edge at vertex v_1, so $\langle v_2, v_1\rangle$ is the N-edge for the vertex v_2, and so on. In general, we have for any integer i:

$\langle v_{4i}, v_{4i+1}\rangle$ is the W-edge for the vertex v_{4i};

$\langle v_{4i+1}, v_{4i+2}\rangle$ is the S-edge for the vertex v_{4i+1};

$\langle v_{4i+2}, v_{4i+3}\rangle$ is the E-edge for the vertex v_{4i+2}; and

$\langle v_{4i+3}, v_{4i+4}\rangle$ is the N-edge for the vertex v_{4i+3}.

This implies immediately that the oriented edge $\langle v_{4h}, v_{4h+1}\rangle$ is the W-edge at vertex v_{4h}, so the oriented edge $\langle v_{4h+1}, v_{4h}\rangle$ is the E-edge at vertex v_{4h+1} under the labeling L_M.

Now let M_F be the mesh obtained from mesh M by the (F,k,h)-operation, that is, by inserting edges $[v_1, v_{4h}]$ and $[v_{4h+1}, v_{4k}]$ and deleting the edges $[v_{4h}, v_{4h+1}]$ and $[v_{4k}, v_1]$. As we discussed above, the mesh M_F is a 4-regular graph. Let the mesh M_F

retain the labels assigned by L_M to all the other oriented edges, and let $\langle v_1, v_{4h} \rangle$ be the E-edge of vertex v_1, $\langle v_{4h}, v_1 \rangle$ the W-edge of vertex v_{4h}, $\langle v_{4h+1}, v_{4k} \rangle$ the E-edge of vertex v_{4h+1}, and $\langle v_{4k}, v_{4h+1} \rangle$ be the W-edge of vertex v_{4k}. Clearly, this is a consistent labeling for the mesh M_F. □

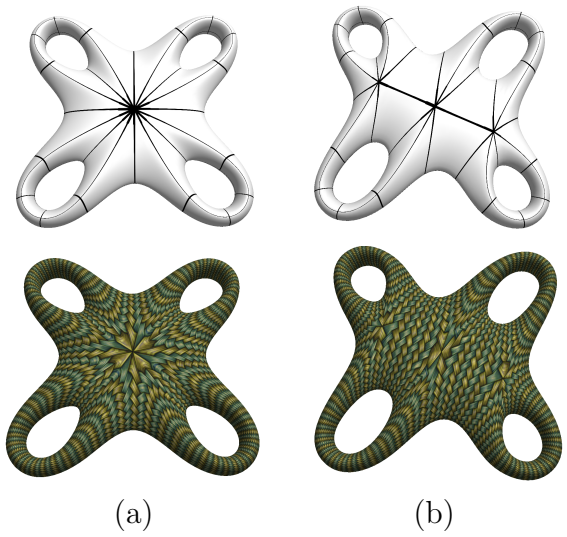


Figure 8.4: Reduction of a 16-valent vertex into three 8-valent vertices, while preserving QPC property.[26]

This provides an effective way to eliminate high valence vertices in QPC meshes, particularly for a mesh with relatively isolated high valence vertices. An example is given in Figure 8.4 to show the reduction operation. In QPC meshes, these operations can reduce vertex valences to release the distortion even at non-saddle points. Here, we do not provide the operation, however It is also possible to move irregular vertices into saddle regions to further reduce the distortions.

8.3 Non-Separating Cycles in Dual of QPC Meshes

A cycle C on a 4-regular mesh is a ***crossing cycle***, if for each vertex v_i of C, the two oriented edges $\langle v_i, v_j\rangle$ and $\langle v_i, v_k\rangle$ on the cycle C are not adjacent in the rotation at v_i. It is easy to verify that each crossing cycle in the dual of a QPC mesh M' is either labeled with N-, S- or E-, W- labels.

Lemma 8.3.1 *Any crossing cycle C in the dual of a QPC mesh M' is non-separating.*

PROOF. Assume a crossing cycle $C = (v_1, e_1, v_2, e_2, \ldots, v_n, e_n, v_1)$ in M' is a separating cycle, where e_i is an oriented edge starting at v_i and ending at v_{i+1}, $1 \leq i \leq n$, $v_{n+1} = v_1$. The crossing cycle C separates the mesh M' into two disjoint parts, M'_l and M'_r, where $M' = M'_l \cup C \cup M'_r$. Considering there are 4 edges incident to each vertex v_i and two of them are on C, the other two edges separated by C are in M'_r and M'_l respectively.

It is safe to assume the oriented edges in C are labeled with N- and S- labels, and C interests with another crossing cycle C' at v_i. Since M'_r and M'_l have no common edge and C' is a cycle, C' should intersect with C at least twice. Let v_1 and v_m be two adjacent intersections on C', P be a path from v_1 to v_m on C and P' be a path from v_m to v_1 on C'. We have a closed walk w which starts from v_1, traverses along P, P' and comes back to v_1. If P' is in $M'_r \cup C$, the corner costs at v_1, v_m are both 3. If P' is in $M'_l \cup C$, the corner costs at v_1, v_m are both 1. Assume there are k_1 edges in P and k_2 edges in P'. The edge-length of w is $k_1 + k_2$, and the corner cost of w is $2(k_1+k_2)-2$ or $2(k_1+k_2)+2$. According to Theorem 8.1.2, twice of edge-length plus corner cost of any closed walk is divisible by 4. However, $2(k_1 + k_2) + 2(k_1 + k_2) \pm 2$ dose not satisfy the condition, which contradicts with the assumption that M' is the dual of a QPC mesh. □

Corollary 8.3.2 *Any crossing cycle C in the dual of a QPC mesh M' is non-contractible.*

A crossing cycle in the dual of QPC mesh is neither contractible and separating. Note that a non-separating cycle can only exist on surfaces with positive genus, which provides another evidence for the fact that a QPC mesh can not be imbedded on a sphere.

9. PERMUTATION VOLTAGE GRAPHS FOR CHILD-TWILLABLE WEAVING AND QUAD-PATTERN COVERABLE MESHES*

The child-twillable weaving introduced in Chapter 7 and the QPC meshes discussed in Chapter 8 have the similar symmetric structures, where a child-twillable weaving consists of two simple patterns and a QPC mesh can be covered with one periodic pattern. In this chapter, we use permutation voltage graph to describe the underlying symmetric structures, where a permutation voltage graph has advantages to represent complicated graph with a simple base graph associated with a permutation assignment[21].

Definition Two imbedded graph $\iota(G_1) = (V_1, E_1)$ and $\iota(G_2) = (V_2, E_2)$ are ***imbedding isomorphic*** if there exists an isomorphism function f between two graphs G_1 and G_2, such that for each vertex $v_i \in V_1$, if the oriented edge (v_j, v_i) is followed by the oriented edge (v'_j, v_i) in the rotation at v_i, the oriented edge $(f(v_j), f(v_i))$ is followed by the oriented edge $(f(v'_j), f(v_i))$ in the rotation at $f(v_i)$.

The formal definition of permutation voltage graph[21] is given as follows.

Definition Let $G = (V, E)$ be a digraph. A ***permutation voltage assignment*** for G is a function $\alpha : E \to X$ that labels each edge with a permutation in the symmetric group X: (1) The pair $\langle G, \alpha \rangle$ is called a ***permutation voltage graph***; (2) Graph G is called the ***base graph*** and group X is called the ***permutation voltage***

*reprinted with permission from "Pattern mapping with quad-pattern-coverable quad-meshes" by Shiyu Hu, Qing Xing, Ergun Akleman, Jianer Chen, Jonathan L. Gross, 2012. Computers & Graphics 36(5): 455-465, Copyright[2012] by Elsevier.

group; (3) The permutation label π_e in α is called the ***voltage*** on edge e.

Definition The ***permutation derived graph*** G^α associated with a permutation voltage graph $(G = (V, E), \alpha : E \to X)$ is defined as follows: (1) $V(G^\alpha) = V \times \{1, \ldots, n\}$, the Cartesian product; (2)$E(G^\alpha) = E \times \{1, \ldots, n\}$; (3) If the edge e is from vertex u to vertex v in G then the edge $e_j = (e, j)$ is from the vertex $u_j = (u, j)$ to the vertex $v_{\alpha(j)} = (v, \alpha(j))$, where $\alpha(\cdot)$ is next element of j on the voltage π_e.

A natural extension is to imbed a base graph G on a surface S by assigning a rotation system to the graph G, denoted as an ***imbedded base graph*** $\iota(G)$. We imbed the corresponding permutation derived graph as an ***imbedded permutation derived graph*** $\iota(G^\alpha)$ with the rotation system inheriting from the imbedded base graph G. Let e_1, e_2 be two oriented edges originating at u in $\iota(G)$, where e_1 is immediately followed by e_2 in the rotation system. In the imbedded permutation derived graph $\iota(G^\alpha)$, (e_1, j) and (e_2, j) are two oriented edges originating at (u, j), where (e_1, j) is immediately followed by (e_1, j) in the rotation at (u, j). The oriented edge (e_1, j) is starting from (u, j) to $(v, \alpha(j))$ if e_1 has the same direction with the underlying directed edge in $\iota(G)$. Otherwise, (e_1, j) goes from (u, j) to $(v, \alpha^{-1}(j))$.

9.1 Permutation Voltage Graph for Child-Twillable Weaving

Recall that a weaving M is a child-twillable weaving, if we can label each cross v as B-type or C-type and the oriented segments originating at v are labeled as North(N-), East(E-), South(S-) and West(W-) in clockwise order, so that any two adjacent crosses v_1 and v_2 satisfy the following conditions: (1) v_1 and v_2 must have different types; and (2) the oriented segments are labeled consistently, for every

oriented segment labeled with (E-), (S-), (W-) or (N-), the oppositely oriented segment is labeled with (W-), (N-), (E-) or (S-).

For a child-twillable weaving σ, the oriented edges of σ-imbedding ι_σ always can be divided into subsets L_1 and L_2, where the oriented edges in L_1 are labeled with (E-) and (W-) labels, and the oriented edges in L_2 are labeled with (N-) and (S-) labels.

9.1.1 Construct child-twillable weaving

We construct an imbedded base graph D_4 with only two 4-valence vertices u and v, where each directed edge is from u to v. We assign u with B-type and v with C-type. We label the oriented edges originating at u and v as (E-), (S-), (W-) and (N-) in clockwise order. For each oriented edge labeled with (E-), (S-), (W-) or (N-), the opposite oriented edge is labeled with (W-), (N-), (E-) or (S-), as shown in Figure 9.1. The imbedded base graph D_4 is the σ-imbedding of a child-twillable weaving. We assign n size permutations $\pi_1, \pi_2, \pi_3, \pi_4$ to the four directed edges originating at u in D_4 respectively to construct an imbedded permutation voltage graph.

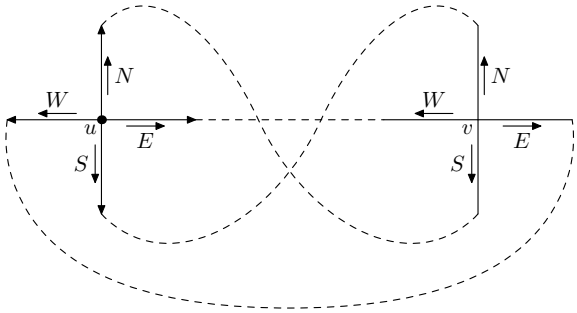


Figure 9.1: An imbedded base graph D_4

Theorem 9.1.1 *The undirected imbedded permutation derived graph D_4^α associated with any permutation voltage assignment α is the σ-imbedding of a child-twillable weaving.*

PROOF. Given an arbitrary permutation voltage assignment $\alpha = \{\pi_1, \pi_2, \pi_3, \pi_4\}$, we assign $\pi_1, \pi_2, \pi_3, \pi_4$ to the four directed edges in D_4, which have the same directions with the four oriented edges labeled with (E-), (S-), (W-) and (N-) originating at u respectively. Let the directed edges derived from $\pi_1, \pi_2, \pi_3, \pi_4$ be E_1, E_2, E_3, E_4 in $\iota(G^\alpha)$. We label the oriented edges along the directed edges in E_1, E_2, E_3, E_4 as (E-), (S-), (W-) or (N-) respectively, and we label their opposite oriented edges as (W-), (N-), (E-) and (S-) respectively. Therefore, for each vertex v' in D_4^α, there are four oriented edges labeled with (E-), (S-), (W-) and (N-) originating at v'. By adding necessary handles, it is easy to make the rotation at each vertex is (E-), (S-), (W-) and (N-) in clockwise order. The vertices derived from u are assigned B-type and the vertices from v are assigned C-type. In the permutation derived graph D_4^α, there is no two adjacent B-type vertices or C type vertices, since there is no self-loops at u or v. According to the definition of child-twillable weaving, the undirected imbedded permutation derived graph is the σ-imbedding of a child-twillable weaving.

□

Then we show the permutation derived graph D_4^α can construct any child-twillable mesh.

Theorem 9.1.2 *For any given child-twillable weaving σ, there exists an permutation voltage assignment α, such that the undirected imbedded permutation derived graph D_4^α is imbedding isomorphic to the σ-imbedding ι_σ.*

PROOF. Suppose ι_σ is the underlying imbedding of a child-twillable weaving σ, we label the vertices in ι_σ with B-type and C-type, where any two adjacent vertices have different types. We first show that the number of vertices with B-types is equal to the number of vertices with C-types. Assume there are n vertices with B-type. Then, there are $4n$ edges in ι_σ. Since two ends of an edge is a B-type vertex and a C-type vertex, the total degree at vertices with C-type is $4n$. Considering each vertex in ι_σ is 4-valence, there are n vertices with C-type. We label all B-type vertices as $\{u_1, u_2, \ldots, u_n\}$ and all C-type vertices as $\{v_1, v_2, \ldots, v_n\}$.

According to Lemma 7.2.1, we can consistently label the oriented edges in ι_σ with (N-), (E-), (S-) and (W-), such that from each vertex v the oriented edges are (N-), (E-), (S-) and (W-) in clockwise order and their opposite oriented edges are (S-), (W-), (N-) and (E-) respectively. We only consider the oriented edges starting from u_i ($i = 1, 2, \ldots, n$). It is known that there are $4n$ such oriented edges. We further separate the $4n$ oriented edges into four group, P_1, P_2, P_3 and P_4, which are labelled with (N-), (E-), (S-) and (W-) respectively. The size of each edge group is exactly n.

We organize the n oriented edges in P_k ($k = 1, 2, 3, 4$) into lists as follows. Repeatedly, we start with an unvisited oriented edge $\langle u_{k_i}, v_{k_j} \rangle$ in P_k, and we continuously add an oriented edge at the end and mark it as *visited* until no next oriented edge is available. The next oriented edge is picked if the subscript of the start vertex of the next oriented edge is the same with the subscript of the end vertex of the last oriented edge in the list. We obtain a cyclic order of subscripts from each edge list $\langle u_{k_i}, v_{k_j} \rangle, \langle u_{k_j}, v_{k_{j'}} \rangle \ldots, \langle u_{k_m}, v_{k_{m'}} \rangle, \langle u_{k'_m}, v_{k_i} \rangle$ as $(i, j, j', \ldots, m, m')$. Since each vertex is shown exactly once in the edge lists of P_k, the orders obtained from the edge lists of P_k constructs disjoint permutation cycles containing numbers from 1 through n.

Therefore, we have a permutation π_k for each set P_k.

We assign the permutations $\alpha = (\pi_1, \pi_2, \pi_3, \pi_4)$ to the directed edges in D_4, which have the same directions with the four oriented edges (N-), (E-), (S-) and (W-) originating at u respectively. We claim the imbedded permutation derived graph D_4^α is imbedding isomorphic to ι_σ. In the imbedded permutation derived graph D_4^α, there are $2n$ vertices, which is equal to the number of vertices in ι_σ. For each vertex u_i' in D_4^α, there are four oriented edges $\langle u_i', v'_{\pi_1(i)}\rangle$, $\langle u_i', v'_{\pi_2(i)}\rangle$ $\langle u_i', v'_{\pi_3(i)}\rangle$ and $\langle u_i', v'_{\pi_4(i)}\rangle$ originating at u_i', which are labeled with (N-), (E)-, (S-) and (W-) respectively. Since π_1 is the permutation of the subscripts of labeled (N-) oriented edges starting at u_i in ι_σ, there exists an oriented edge $\langle u_i, v_{\pi_1(i)}\rangle$ labeled with (N-) in ι_σ. Similarly, there are oriented edges $\langle u_i, v_{\pi_2(i)}\rangle$ $\langle u_i, v_{\pi_3(i)}\rangle$ and $\langle u_i, v_{\pi_4(i)}\rangle$ in ι_σ. On the other hand, for each oriented edge $\langle u_i, v_j\rangle$ in ι_σ, j is immediately following i in π_1, π_2, π_3 or π_4, which induces an oriented edge $\langle u_i', v_j'\rangle$. □

Theorem 9.1.3 *For the σ-imbedding of a child-twillable weaving σ, there is an one-to-one mapping between a consistent labeling L and a permutation voltage assignment $\alpha = (\pi_1, \pi_2, \pi_3, \pi_4)$ of D_4, where $\pi_1, \pi_2, \pi_3, \pi_4$ are assigned to the four directed edges which are originating at u and labeled as (N-), (E-), (S-) and (W-) in clockwise order. The undirected D_4^α is imbedding isomorphic to ι_σ and the derived labeling of D_4^α is the same with L.*

9.1.2 Child-twillable mesh on any positive genus

According to Corollary 7.2.6, we already know that a child-twillable weaving only exists on a surface with positive genus. In the following part, we provide a method to construct the σ-imbedding of a child-twillable weaving on any surface with positive genus.

Lemma 9.1.4 *For an integer $g \geq 1$, there exists a permutation voltage assignment α for D_4, such that the imbedded permutation derived graph D_4^α is cellularly imbedded on a surface with genus g.*

PROOF. Given an integer $g \geq 1$, we design the permutation voltage assignment $\alpha = (\pi_1, \pi_2, \pi_3, \pi_4)$, where $\pi_1 = (1, 2, \ldots, g)$, $\pi_2, \pi_3, \pi_4 = (1)\ (2)\ \ldots\ (g)$. We assign $\pi_1, \pi_2, \pi_3, \pi_4$ to the four directed edges of D_4, which have the same directions with the oriented edge originating at u labeled with (E-), (S-), (W-) and (N-) respectively.

In the imbedded permutation derived graph D_4^α, there are $2g$ vertices $u_1, u_2, \ldots, u_g$ and $v_1, v_2, \ldots, v_g$. We consistently label the oriented edges of D_4^α with (E-), (S-), (W-) and (N-), such that the oriented edges, which share the same directions with the directed edges derived from π_1, π_2, π_3 and π_4, are labeled with (E-), (S-), (W-) and (N-) respectively, and their opposite oriented edges are labeled with (W-), (N-), (E-) and (S-). The (E-) oriented edges, which are along the directed edges derived from permutation π_1, are $P_1 = \{\langle u_1, v_2\rangle, \langle u_2, v_3\rangle, \ldots \langle u_{g-1}, v_g\rangle, \langle u_g, v_1\rangle\}$. The other ($E$-) oriented edges, which are opposite the directed edges derived from π_3, are $P_3^{-1} = \{\langle v_1, u_1\rangle, \langle v_2, u_2\rangle, \ldots \langle v_{g-1}, u_{g-1}\rangle, \langle v_g, u_g\rangle\}$. We combine the two groups of oriented edges P_1 and P_3^{-1} into a Hamilton circle $H = (u_1 \to v_2 \to u_2 \to v_3 \to \cdots \to u_g \to v_1 \to u_1)$. Similarly, we consider (S-) oriented edges P_2, P_4^{-1} derived from π_2 and π_4^{-1}. Combine P_2 and P_4^{-1}, we construct g cycles T, $(u_1 \to v_1 \to u_1)$, $(u_2 \to v_2 \to u_2)$, $\ldots$, $(u_g \to v_g \to u_g)$.

Starting from the Hamilton circle H, we one-by-one insert the g cycles with consistently labeling, where after inserting all cycles the oriented edges at a vertex are labeled with (E-), (S-), (W-) and (N-) in clockwise order and the opposite oriented edges are (W-), (N-), (E-) and (S-). When we insert a cycle of T to H, the genus is increased by 1, since only the number of edges is increased by 2. It is easy

to realize the imbedded graph after inserting all cycles is exactly D_4^{α}. □

9.2 Permutation Voltage Graph for QPC Meshes

It is shown that a QPC mesh can be seamlessly cover a periodic quadrangular pattern. The necessary and sufficient conditions to identify a QPC mesh are also given in Theorem 8.1.2. In this section, we show a QPC mesh can be represented by a simple imbedded permutation voltage graph and we provide a method to construct a QPC mesh on any surface with positive genus.

9.2.1 Construct QPC meshes

The global symmetric appearances of QPC meshes with local consistent labeling enjoys the similar property of the underlying imbedding of a child-twillable waving. Based on this observation, we try to represent the dual of QPC meshes as an imbedded permutation derived graphs. We start with the construction of the base graph.

We construct a bouquet B_2 with only one vertex v and two directed self-loops c_1, c_2 (see Figure 9.2), and the bouquet B_2 can be cellularly imbedded on a torus. The four oriented edges originating at v are labeled with (E-), (S-), (W-) and (N-) in clockwise order, where (E-) and (W-) share the same circle c_1, and (S-) and (N-) share the same circle c_2. The oriented edges labeled with (E-) and (N-) have the same directions along c_1, c_2 respectively. We assign a permutation voltage assignment $\alpha = (\pi_1, \pi_2)$ to the two directed edges c_1, c_2, where π_1, π_2 are permutations with n elements. Each directed edge in B_2 derives n directed edges in B_2^{α}. The oriented edges, which have the same directions along the directed edges derived from c_1, c_2, are labeled with (E-), (N-) respectively.

Lemma 9.2.1 *Given the dual of a QPC mesh M', the edges in M' always can be*

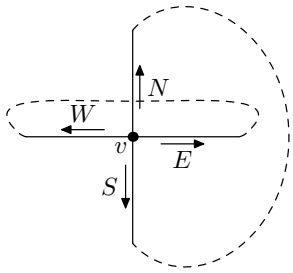


Figure 9.2: A imbedded bouquet B_2 with two self-circles

divided into two spanning subgraphs L_1 and L_2, where the oriented edges in L_1 are labeled with (E-) and (W-) labels, and the oriented edges in L_2 are labeled with (N-) and (S-) labels.

PROOF. According to the definition of the dual of QPC meshes, each vertex v in the dual of a QPC mesh M' has 4 oriented edges originating at v, where (E-), (S-), (W-) and (N-) oriented edges in clockwise order. The common edges between any two adjacent vertices are either labeled with (E-) and (W-) or (N-) and (S-). We collect the edges labeled with (E-), (W-) into set L_1, and the edges labeled with (N-), (S-) into set L_2. Since there are oriented edges labeled with (E-) and (N-) staring at any vertex v, L_1 and L_2 are both spanning subgraphs. □

In the following, we show the imbedded permutation voltage graph associated with any permutation voltage assignment for B_2 induces the dual of a QPC mesh. According to Lemma 9.2.1, the dual of a QPC mesh can be divided into two spanning subgraphs. It is obvious that the edges in the imbedded permutation derived graph B_2^{α} can be also divided into two subsets T_1 and T_2, where the oriented edges in T_1 are labeled with (E-), (W-), which are derived from the self-circle c_1, and the others in T_2 are labeled with (N-) and (S-), derived from the self-circle c_2. T_1, T_2 are spanning

subgraphs in B_2^α.

Theorem 9.2.2 *Given the imbedded base graph B_2, where we assign a permutation voltage assignment $\alpha = (\pi_1, \pi_2)$ to the two directed edges of B_2, the undirected imbedded permutation derived graph B_2^α is the dual of a QPC mesh.*

PROOF. According to the construction of B_2, the four oriented edges of B_2 are labeled with (E-), (S-), (W-) and (N-) in a clockwise order at the vertex v. The two oriented edges (E-) and (N-) are along the same directions with the two directed edges c_1, c_2 respectively. In the imbedded permutation derived graph B_2^α, we label the oriented edges which are along the directed edges derived from c_1 and c_2 as (E-) and (N-) respectively, and we label their opposite oriented edges as (W-) and (S-) respectively. Therefore, the oriented edges labeled with (E-) and (W-) share edges; the oriented edges labeled with (N-) and (S-) share edges in B_2^α. For each vertex v' in B_2^α, there are four oriented edges originating at v', which are labeled with (E-), (S-), (W-) and (N-). Since B_2^α inherits the rotation system from B_2, the four oriented edges (E-), (S-), (W-) and (N-) are in a clockwise order at each vertex in B_2^α. Thus, the imbedded permutation derived graph B_2^α has a consistent labeling. According to the definition, the undirected B_2^α is the dual of a QPC mesh. □

Theorem 9.2.2 shows that the imbedded permutation derived graphs B_2^α are a subset of the dual of QPC meshes. We show below that for the dual of any QPC mesh, we can obtain a permutation voltage assignment such that the corresponding imbedded derived graph B_2^α is the dual of the given QPC mesh, which completes the two-way mapping.

Theorem 9.2.3 *There exists an undirected imbedded permutation derived graph B_2^α, which is imbedding isomorphic to the dual of a given QPC mesh M'.*

PROOF. Assume there are n vertices in M', which are $v_1, v_2, \ldots, v_n$. According to Lemma 9.2.1, we can divide M' into two spanning subgraphs L_1, L_2, where oriented edges in L_1 are labeled with (E-), (W-) and the oriented edges in L_2 are labeled with (N-), (S-). We assign directions to the undirected edges in L_1, L_2, such that the edges in L_1 have directions along the (E-) oriented edges and the edges in L_2 have directions along the (N-) oriented edges. We traverse L_1, L_2 along the directed edges. If L_i ($i = 1, 2$) is a connected subgraph, which is also a Hamiltonian cycle, we obtain a cyclic permutation π_i corresponding to the order of subscripts of the vertices encountered in L_i. If L_i contains several disjoint cycles $c_1, c_2, \ldots, c_m$, we can obtain a cyclic permutation for each cycle and the permutation π_i is the product of the disjoint cycle permutations. Therefore, we have a permutation voltage assignment $\alpha = \{\pi_1, \pi_2\}$.

We assign the permutation $\alpha = \{\pi_1, \pi_2\}$ to the two directed edges c_1, c_2 in B_2, which share the same directions with (E-) and (N-) oriented edges. Since there are n elements in the permutations π_1, π_2, there are n vertices in the imbedded permutation derived graph B_2^α, which are denoted as $v_1', v_2', \ldots, v_n'$. For the directed edges derived from c_1(resp. c_2), we label the oriented edges, which share the same directions with the directed edges, with (E-) (resp. (N-)), and we label their opposite oriented edges as (W-) (resp. (S-)). According to Theorem 9.2.2, we know undirected B_2^α is the dual of a QPC mesh. In the following, we show that the undirected B_2^α is imbedding isomorphic to M'. We build an one-to-one mapping between the vertices in M' and B_2^α, such that v_i is mapped to v_i' for $1 \leq i \leq n$. Without loss of generality, assume $e = (v_i, v_j)$ is an edge labeled with (E-) and (W-) in M', where e is from v_i to v_j which has the same direction with the (E-) oriented edge. In our construction, i is immediately ahead j in the permutation π_1. Since we assign π_1 to c_1, in the imbedded permutation derived graph, there is a directed edge from v_i' to v_j' which share the

same direction with a (E-) oriented edge in B_2^α. On the other hand, an directed edge from v_i' to v_j' in B_2^α, which has the same direction with (E-) oriented edges, indicates i is immediately ahead of j in π_1, which further indicates an directed edge from v_i to v_j in L_1. Therefore, according to the definition of imbedding isomorphic, the undirected B_2^α is imbedding isomorphic to M'. □

Theorem 9.2.4 *For the dual of a QPC mesh M', there is an one-to-one mapping between a consistent labeling L and a permutation voltage assignment $\alpha = (\pi_1, \pi_2)$ of B_2, where π_1, π_2 are assigned to the two directed edges which are originating at v and labeled as (E-), (N-). The undirected B_2^α is imbedding isomorphic to ι_σ and the derived labeling of B_2^α is the same with L.*

9.2.2 QPC meshes on any positive genus surface

According to Lemma 8.1.7, it is known that QPC meshes can only exists on surfaces with positive genus. In the following, we construct the dual of QPC meshes on any surfaces with positive genus.

Theorem 9.2.5 *For any integer $g \geq 1$, there exists an imbedded permutation derived graph B_2^α on the surface with genus g, where the undirected B_2^α is the dual of a QPC mesh M'.*

Proof. Fix the integer $g \geq 1$. We design a permutation voltage assignment $\alpha = (\pi_1, \pi_2)$, where $\pi_1 = (1\ 2\ \ldots\ 2g-1\ 2g)$ and $\pi_2 = (1\ 2)(3\ 4)\ldots(2g-1\ 2g)$. We label the four oriented edges at the vertex v in B_2 with (E-), (S-), (W-) and (N-) in a clockwise order, such that (E-), (W-) oriented edges share the same edge and (N-), (S-) share the same edge. We assign π_1, π_2 to the two directed self-loops c_1, c_2 in B_2, where the oriented edges labeled with (E-) (N-) share the same directions

with c_1, c_2 respectively. In the imbedded permutation derived graph B_2^{α}, we label the oriented edges, which share the same directions with the directed edges derived from c_1(resp. c_2), with (E-) (resp. (N-)).

The directed edges derived from c_1 sequentially connect the vertices from v_1 to v_{2g} as a Hamilton circle. The directed edges derived from c_2 construct g circles, where each circle contains two consecutive vertices v_{2i-1} and v_{2i} ($1 \leq i \leq g$). By adding necessary handles, it is easy to make (E-), (S-), (W-) and (N-) oriented edges originating at each vertex are in clockwise order. Then we show the genus of B_2^{α} is g.

We draw a circle C on a plane, and put $2g$ vertices on the circle, which are labeled as $v_1, v_2, \ldots, v_{2g}$ sequentially in a clockwise order. There are $2g$ vertices, $2g$ edges and 2 faces in the circle C. The oriented edges on the circle C are labeled with (E-) and (W-), where the oriented edges from v_j to v_{j+1} ($1 \leq j \leq 2g, v_{2g+1} = v_1$) are labeled with ($E$-). One by one, we insert the g cycles into C. For each cycle ($v_{2i-1} \to v_{2i} \to v_{2i-1}$), we label the oriented edge from v_{2i-1} to v_{2i} and the oriented edge from v_{2i} to v_{2i-1} with (N-), and their opposite oriented edges are labeled with (S-). Therefore, by making the labeling consistent, the genus is increased by 1 when we insert a circle $v_{2i-1} \to v_{2i} \to v_{2i-1}$, since only the number of edges is increased by 2. □

From the proof above, we present a method to create the dual of a QPC mesh on surfaces with any positive genus.

In this chapter, we use imbedded permutation voltage graphs to construct the underlying imbedded graph of child-twillable weaving and the dual of a QPC mesh with simple imbedded base graphs B_2 and D_4 respectively. On the other hand, any undirected imbedded permutation derived graph based on B_2 and D_4 is the

underlying imbedded graph of a child-twillable weaving and the dual of a QPC mesh respectively. The one-to-one mapping between a permutation voltage assignment of B_2(resp. D_4) and a consistent labeling of the dual of a QPC mesh(resp. underlying imbedded graph of a child-twillable weaving) provides a simple way to represent them.

10. SOFTWARE DESIGN: PLANE WEAVING CONSTRUCTION AND RECONSTRUCTION

In this chapter, we present an application of the weaving theory in practice. We build a software ***WvConstructor*** which can construct plane weavings based on plane graph drawings. Moreover, with a high-quality image of weaving as input, we use the WvConstructor to analyze the underlying planer graph and reconstruct the weaving, which would be available for modification, saving and transmitting. Since all the theorems used in this software are already given in the previous chapters, we only describe the functions and show some results.

10.1 Overall Description

WvConstructor is developed with C++11 (gcc 4.6.2) under Ubuntu 11.04. The User Interface(UI) part is powered by FLTK, and the other() involved library tools are OpenCV and OpenGL.

The main panel of WvConstructor contains seven parts, as shown in Figure 10.1.

1. Drawing and Display Area: User can draw planar graphs(connected or disconnected) in this area, and the induced weaving is also displayed here.

2. Graph Drawing: After drawing a graph, user can use the provided functions to edit the vertices, edges, twisted types and control points.

3. Save & Load Drawing: User can use this function to save a drawing into a hard disc or load an existing drawing.

4. Weaving Reconstruction: User can load an image of weaving into WvConstructor to obtain the underlying planar graph and reconstruct the weaving based

on the detected planar graph.

5. Adjust Appearance of Input Image: A loaded weaving image would be shown as background in the Display area, which is useful for the weaving reconstruction, especially when the reconstruction algorithm cannot rebuild the weaving correctly.

6. Display Option: User can select interested parts, like underlying graph or induced weaving, to review.

7. Reset: Reset WvConstructor into the initial setting. Clear the loaded image and user drawing.

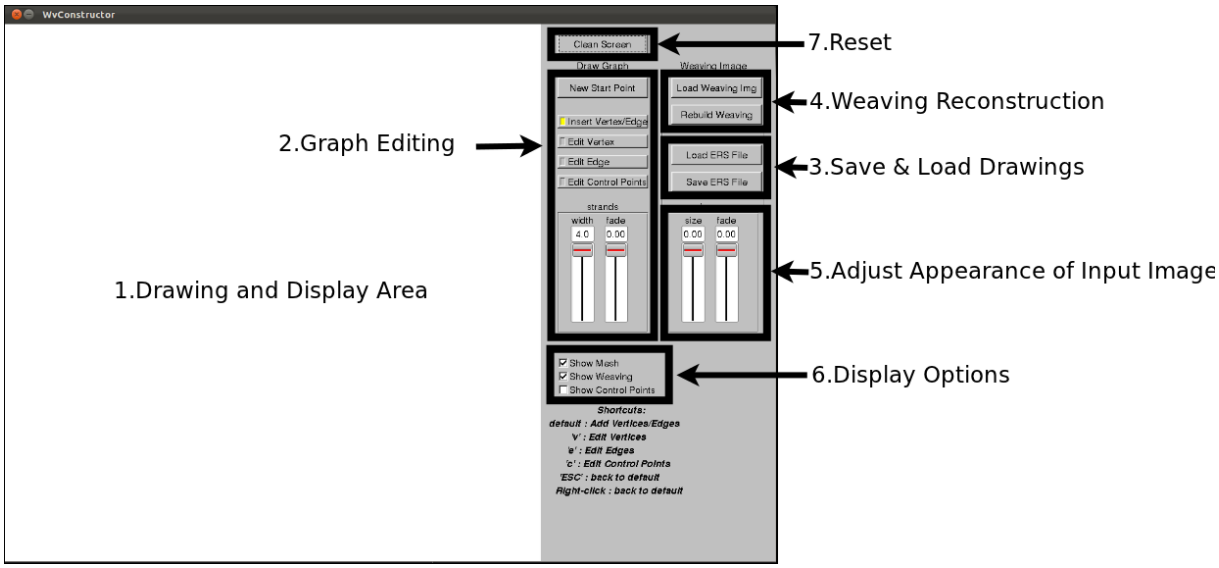


Figure 10.1: Main panel of WvConstructor

10.2 Planar Graph Drawing and Induced Plane Weaving

In this section, we introduce one of the main user cases of WvConstructor, planar graph drawing and editing. With the user's drawing and editing, WvConstructor updates the induced plane weaving automatically.

WvConstructor uses four types of status to support graph drawing and editing, ***Insert Vertex and Edge***, ***Vertex Editing***, ***Edge Editing*** and ***Control Points Editing***.

10.2.1 Insert vertex and edge

Under this status, the last touched vertex is set as the ***active*** vertex, which is important for WvConstructor to create new edges. WvConstructor would draw a new vertex in an empty drawing area where a mouse left-clicked, and set the new vertex as current active vertex. WvConstructor would react an existing vertex with an right-click on it. By clicking the ***new start point*** button or clicking the right-click button of the mouse, WvConstructor would clear the active vertex, which is used to draw a disconnected graph.

There are three scenarios supported by WvConstructor, as shown in Figure 10.2.

1. *Insert a new vertex without new edge.* Without active vertex, left-click in an empty drawing area would insert a new vertex without creating any new edge.

2. *Insert a vertex with a new edge.* With active vertex, left-click in an empty drawing area would insert a new vertex and insert a new edge between the new vertex and the active vertex. Then, the new created vertex is set as active vertex for the next step.

3. *Insert a new edge between two existing vertices.* With active vertex v_1 existing,

left-click on another existing vertex v_2, WvConstructor would insert a new edge between v_1, v_2, if no such edge exists.

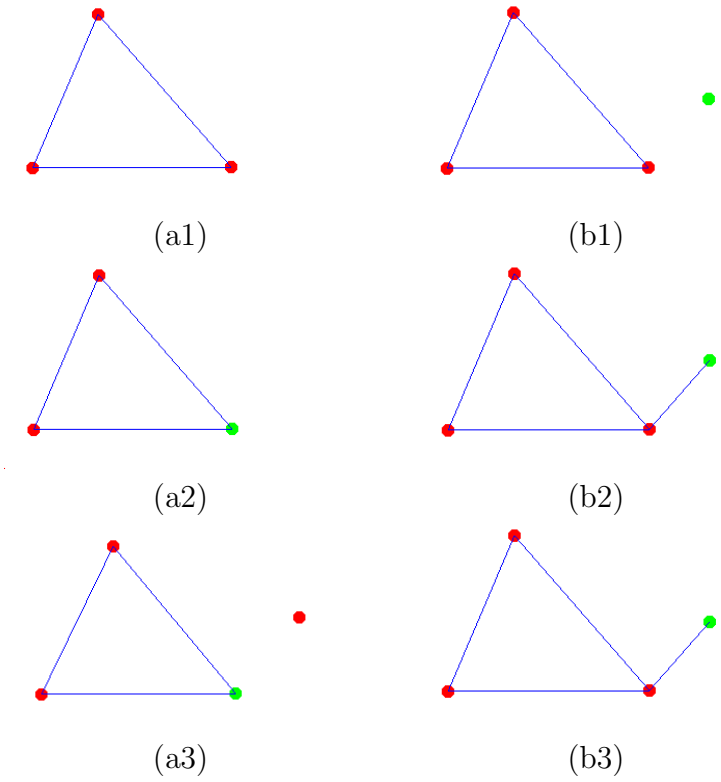


Figure 10.2: Insert vertices and edges (vertex with green color is an active vertex) (a1-b1) Insert a new vertex; (a2-b2) Insert a new vertex and a new edge between the new vertex and active vertex; (a3-b3) Insert a new edge between active vertex and an existing vertex.

10.2.2 Vertex editing

Press "*V*" or press the "*Edit Vertex*" button, WvConstructor would be switched into *Vertex Editing* status. Under this status, WvConstructor supports two types of operations, vertex relocation and deletion, as shown in Figure 10.3.

1. *Vertex Relocation*: Move the mouse onto a chosen vertex, push left-button and drew the vertex to a new position. The rotation system will be recomputed after relocating a vertex.

2. *Vertex Deletion*: Right-click on a chosen vertex, and then click "*delete*" option in the pop-out window to delete the vertex. The edges incident to the deleted vertex will be deleted automatically.

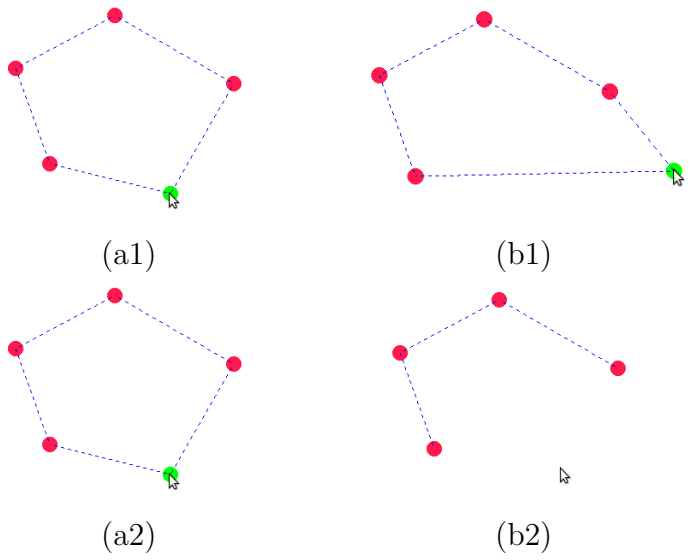


Figure 10.3: Vertex Relocation and Deletion. (a1-b1)Draw a vertex to a new position; (a2-b2) Delete the chosen vertex

10.2.3 Edge editing

Press "*E*" or press the "*Edit Edge*" button, WvConstructor is set as *Edge Editing* status. In Edge Editing status, WvConstructor supports *Edge Contraction*, *Edge Deletion* and *Changing Twisted-type*. *Edge Contraction* and *Edge Deletion* are shown in Figure 10.4, and *Changing Twisted-type* will be introduced later.

1. *Edge Contraction*:Right-click on a chosen edge, and then click "*contract*" option in the pop-out window to contract the edge. After contracting an edge e, the two endpoints v_1, v_2 will merge as one vertex which is located at the middle of the edge e. Meanwhile, the rotation system of the graph would be recomputed.

2. *Edge Deletion*: Right-click on a chosen edge, and then click "*delete*" option in the pop-out window to delete the edge. If edge e is the only incident edge to an endpoint, the endpoint would also be deleted after we deleting e. The rotation system would be recomputed.

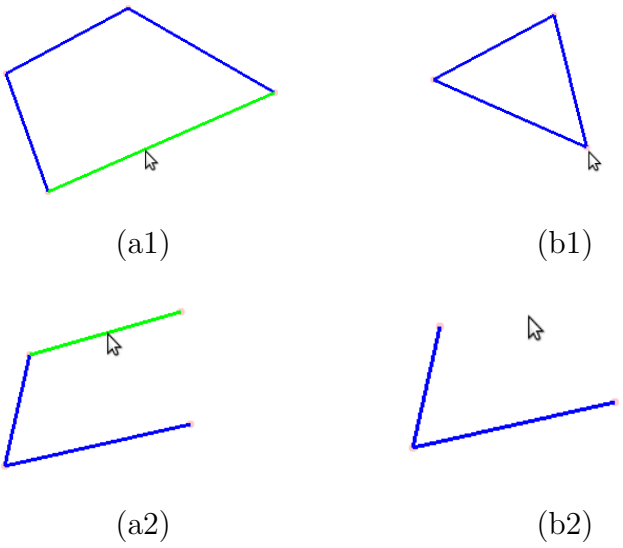


Figure 10.4: Edge Contraction, Edge Deletion and Changing Twist-type. (a1-b1)Contract an edge; (a2-b2) Delete an edge; (a3-d3) Changing the twisted-type of the bottom edge.

10.2.4 Plane weaving construction

During the planar graph drawing and editing process, the induced weaving is updated automatically. The weaving curves in WvConstructor are spliced with Bezier curves of order 4, where each Bezier curve is defined by a face-corner.

When we create a new edge $e = (v_1, v_2)$, we create five control points c_1, c_2, c_3, c_4, c_5 attached to e. The control point c_1 is in the middle of e, and if we draw virtual edges connecting c_2, c_3, c_4, c_5 with c_1, the edges $(v_1, c_1), (c_2, c_1), (c_3, c_1), (v_2, c_1), (c_4, c_1)$ and (c_5, c_1) are in clockwise order oriented at c_1 and the virtual edges $(c_2, c_1), (c_3, c_1), (c_4, c_1)$ and (c_5, c_1) are at the angle of 45° with e respectively, as shown in Figure 10.5. The lengths of virtual edges $(c_2, c_1), (c_5, c_1)$ are proportional to the length of (v_1, c_1), and similarly the lengths of $(c_3, c_1), (c_4, c_1)$ are proportional to the length of (v_2, c_1). Therefore, when an edge is drawn, the positions of the five attached control points are fixed.

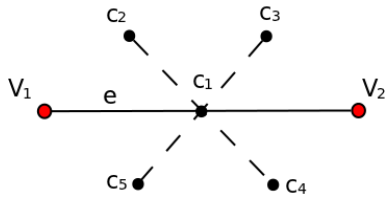


Figure 10.5: Five control points attached to each edge

Let (v, e_1, e_2) be a face corner in the plane graph $\pi(G)$, where e_1, e_2 are oriented edges starting at v, and e_2 is 0-next to e_1 at v. Assume the 5 control points attached to e_1 are c_1, c_2, c_3, c_4, c_5, and the 5 control points of e_2 are $c'_1, c'_2, c'_3, c'_4, c'_5$. WvConstructor uses c_1, c_5, c'_2, c'_1 as control points to define a Bezier curve for the face corner (v, e_1, e_2), as shown in Figure 10.6. In other words, for the edge e_1, $\{c_2, c_1\}, \{c_3, c_1\}, \{c_4, c_1\}$

and $\{c_5, c_1\}$ contribute to the four incident? face corners of e_1, where $\{c_2, c_1\}$ and $\{c_5, c_1\}$ (resp. $\{c_3, c_1\}$ and $\{c_4, c_1\}$) may contribute to the same face corner.

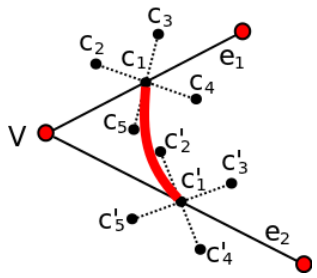


Figure 10.6: A weaving curve defined by four control points $\{c_1, c_5, c_2', c_1'\}$ for face corner (v, e_1, e_2).

Note that each new edge is set as 1^+-twisted type by default in WvConstructor. We give an example of how the weaving appearances changes with a sequence of operations, as shown in Figure 10.7.

With a fixed plane graph, we can still modify the appearances of the weaving by adjusting the width of strands, editing the control points and changing the twisted-types.

10.2.5 Adjusting the width of strands

In WvConstructor, all strands have the same width, which has a default value when WvConstructor starts. We can change the width of the strands by using the slider bar "*width*" in panel "*strand*" of WvConstructor, as shown in Figure 10.8

10.2.6 Relocating control points

The default positions of control points can not satisfy all the requirements for different weaving appearances. WvConstructor provides a function to allow users to relocate the control points. However, WvConstructor does not allow users to add more control points to define a Bezier curve, which is a limitation for the weaving appearance.

Press "c" or push the "*Edit control points*" button, WvConstructor is in the *Control Points Editing* status, where the relocation of control points is available.

To keep weaving strands smooth, WvConstructor sets some rules for the control points relocation operation. Let $e = (v_1, v_2)$ be an edge, where c_1, c_2, c_3, c_4, c_5 are the control points attached to it. By default, the control points $\{c_1, c_2, c_4\}$ and $\{c_1, c_3, c_5\}$ are in the same line respectively, as shown in Figure 10.9(1). When we relocate any control points, WvConstructor would relocate the corresponding control points to keep $\{c_1, c_2, c_4\}$ and $\{c_1, c_3, c_5\}$ still in line, as shown in Figure 10.9(2,3).

10.2.7 Changing twisted types

In "*Edit Edge*" status, left-click on an edge, WvConstructor would change the twisted-type of it. There are four types of twistedtypes in WvConstructor, 1^+-twisted, 1^--twisted, and two types of NULL-twisted, as shown in Figure 10.10. The four twisted-types revolve when users continuously change the twisted-type.

10.3 Weaving Recognition and Reconstruction

In this section, we introduce the *Weaving Reconstruction* module of WvConstructor. By loading an image of weaving, WvConstructor can detect the weaving contours, and reconstruct the underlying plane graph and its induced weaving. The Weaving Reconstruction function would be useful for artists and mathematicians to understand and modify existing weavings only in images.

Since the quality of a reconstructed weaving greatly relies on the quality of the input image, and the purpose of WvConstructor is not image processing, we provide some intuitive requirements for the input images.

1. High Resolution: The resolution is at least 800×600.
2. White and Clean Background.
3. Dark Strands: The color of strands should be in high contrast with the white background. The texture patterns are allowed inside the strands, but the colors should also be in high contrast with the white background.
4. Clear Boundaries: The boundaries between strands and background should be clean and clear.
5. Uniform Strands: In particular, the widths of the strands should be the same.

The main pipeline of the Weaving Reconstruction includes *Image Binarization*, *Contour Detection*, *Gaps Coloring* and *Graph Construction*, as shown in Figure 10.11.

10.3.1 Image binarization

Based on our requirements for the input images, we can adapt simple color threshold in order to convert the input images into binary images, as shown in Figure 10.12.

10.3.2 Contour detection

WvConstrutor uses library OpenCV to handle the contours detection. Every contour detected from the image bounds a gap of the original weaving, as shown in Figure 10.13.

10.3.3 Gaps coloring

As mentioned above, each gap of the original weaving σ is bounded by a detected contour. According to Theorem 5.1.2, the gaps of a plane weaving with connected imbedding is 2-colorable, where one color corresponds to V-gaps and the other corresponds to F-gaps. To restore the plane graph G, whose induced weaving is σ, we need to classify all the gaps into V-gaps and F-gaps correctly. WvConstructor adapts a heuristic algorithm in order to classify all the gaps.

The algorithm *ClassificationVoting* is given in Figure 10.14. The idea of this algorithm is based on the fact that no adjacent gaps are labeled the same. Therefore, for each labeled gap p_i, the adjacent unlabeled gaps could obtain a vote for the opposite type of p_i, and an unlabeled gap can be labeled by the majority of the received votes in the end. During the proof of Theorem 5.1.3, the unbounded gap of a plane weaving always corresponds to a F-gap. WvConstructor starts from the exterior gap, uses Breadth-First-Search and voting algorithms to do the classification. Figure 10.15 gives an example about how WvConstructor identifies the adjacent contours.

10.3.4 Graph construction

After labeling the gaps with V-gaps and F-gaps, WvConstructor creates vertices in the geometric centers of the V-gaps. The last step is to add edges between the new created vertices to construct graphs. The algorithm is simple, as shown in

Figure 10.16. Figure 10.17 gives an example of graph reconstruction and its induced weaving by WvConstrutor.

More weaving reconstruction examples are given in Figure 10.18, 10.19, 10.20, 10.21 and 10.22.

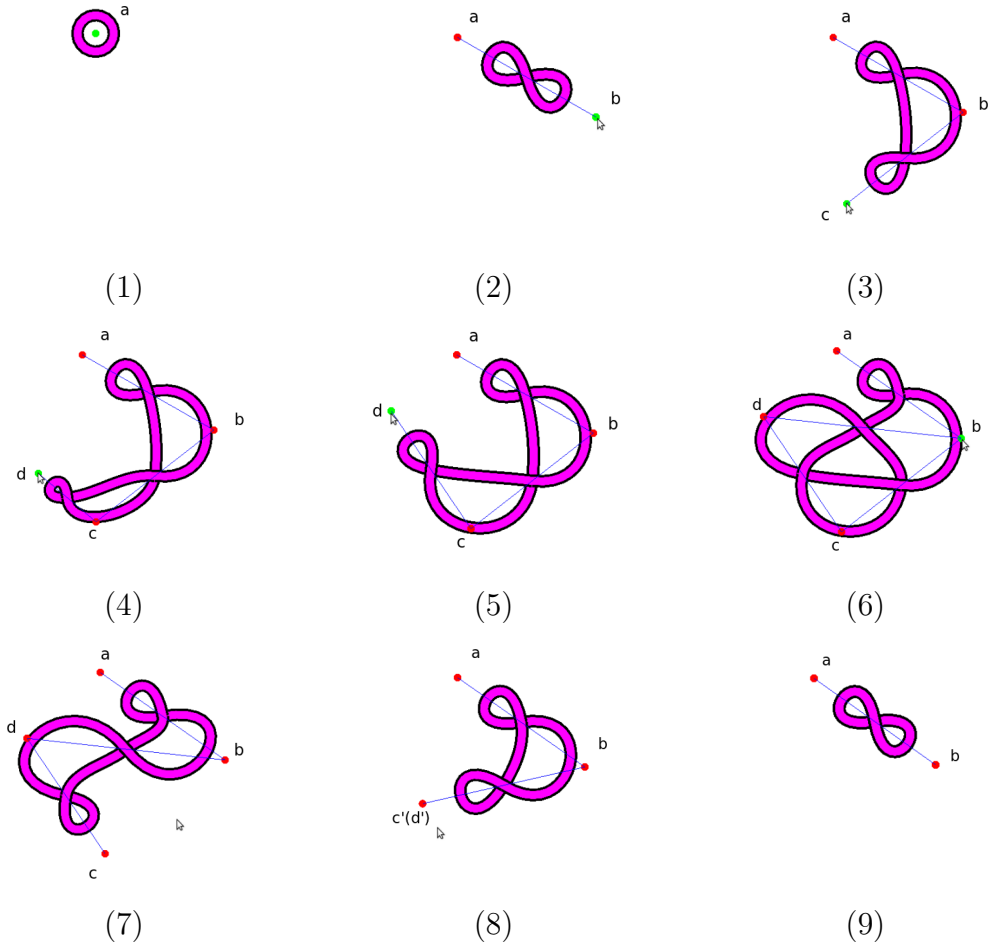


Figure 10.7: A sequence of drawing operations and their updated induced weaving. (1) create a single vertex a; (2) create a new vertex b and a new edge (a, b); (3) create a new vertex c and a new edge (b, c); (4) create a new vertex d and a new edge (d, c); (5) drew vertex d to a new position; (6) create a new edge (d, b); (7) delete edge (b, c); (8) contract edge (d, c) into a vertex $c'(d')$; (9) delete vertex $c'(d')$

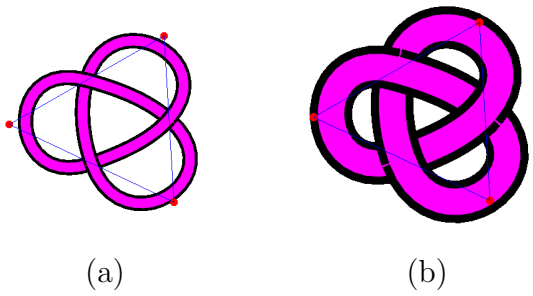


Figure 10.8: Adjusting the width of strands

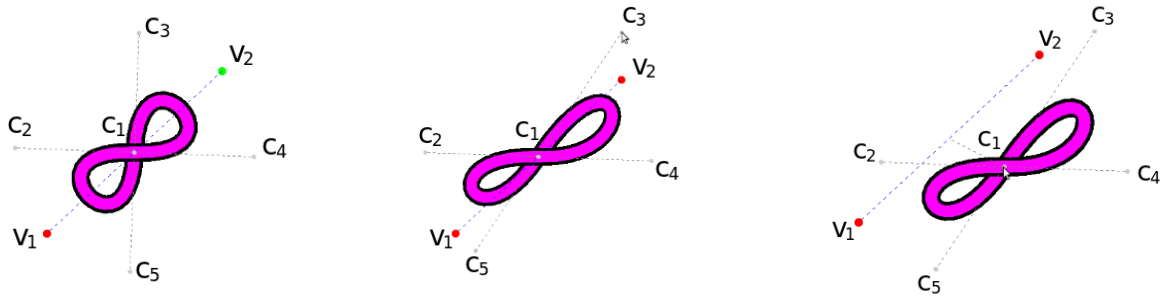


Figure 10.9: A sequence of control points relocation operations and the appearance of weaving. (1) Control points in default positions; (2) Relocate c_3, the position of c_5 is also changed; (3) Relocate c_1, all the other four control points are also relocated.

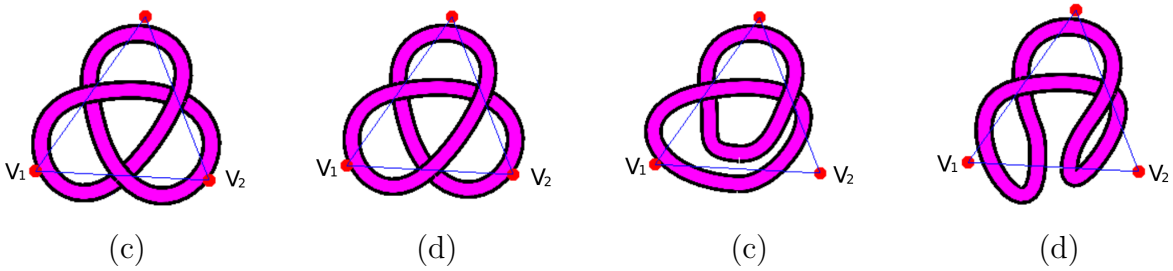


Figure 10.10: The bottom edge (v_1, v_2) with four types of twisted-types in WvConstructor, 1^+-twisted, 1^--twisted, and two types of NULL-twisted

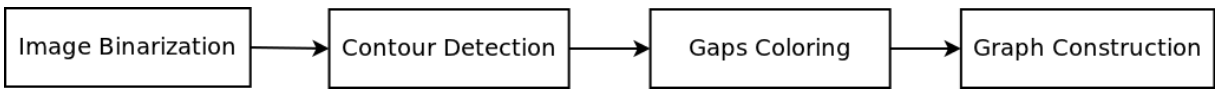
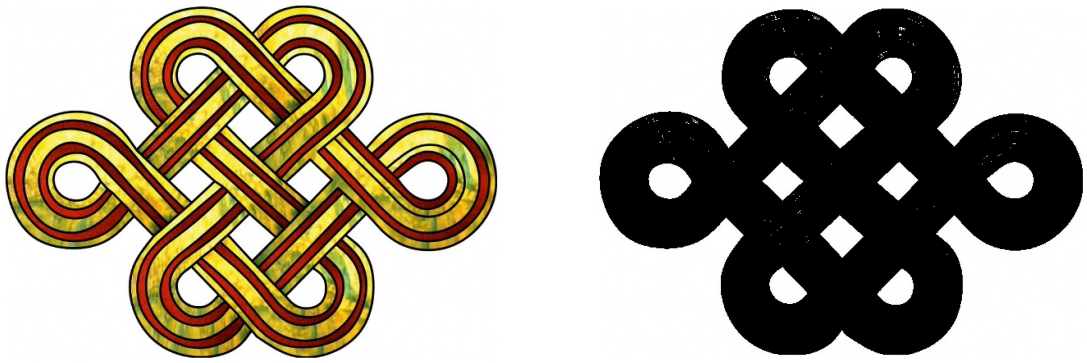
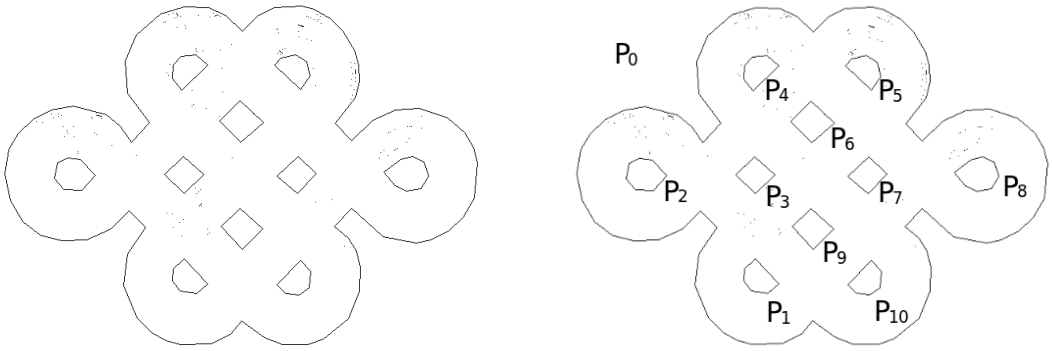


Figure 10.11: The pipeline of Weaving Reconstruction



(a) (b)

Figure 10.12: An example for image binarization. (a) Input image; (b) Binary image



(a) (b)

Figure 10.13: An example of weaving contours detection. (a) The detected contours; (b) Each contour defines a gap, p_0 is the unbounded exterior gap.

ClassificationVoting
INPUT: an array A of detected contours, where a contour is a sequence of 2D points.
OUTPUT: a labeling for contours, either F-gap or V-gap.

```
1   Generate an empty queue Q of contours;
2   Label t_0 with F-gap, where t_0 bounds the exterior gap p_0;
3   Push the contour t_0 into Q;
4   while the queue Q is not empty,
4.1     Pop out the first contour t_i from Q;
4.2     if the F-gap votes of t_i is more than V-gap votes,
4.3         label t_i as F-gap
4.4     else label t_i as V-gap
4.5     for each edge e=(x_i, x_{i+1}) in t_i
4.6         Send out perpendicular rays starting from x_i, x_{i+1} and the middle point
4.7         if the first hit contour t' is not in Q and not labeled
4.8             Push t' into Q
4.9             vote t' as F-gap, if t_i is V-gap; otherwise, vote t' as V-gap;
```

Figure 10.14: Algorithm for classify gaps into F-gap and V-gap

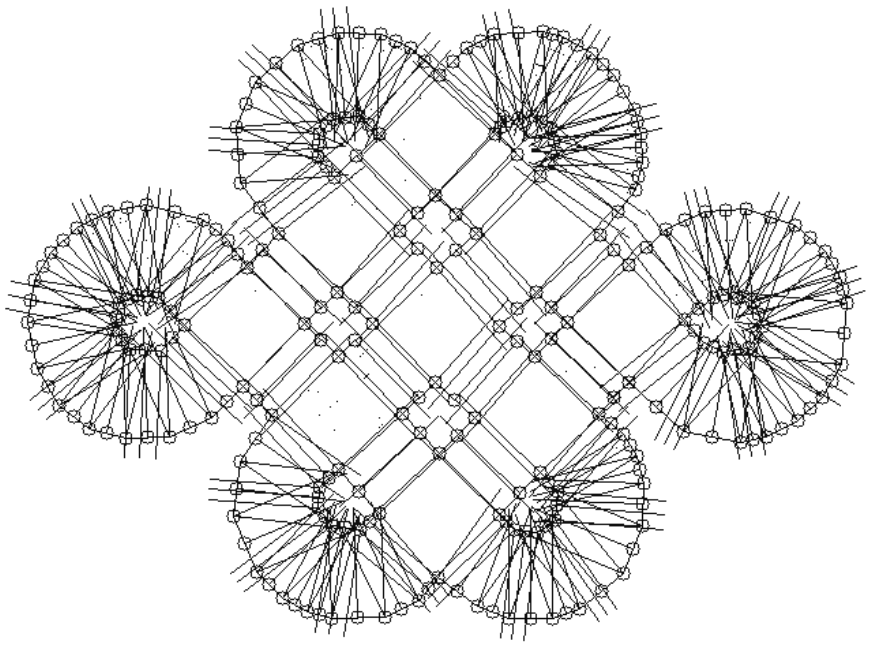
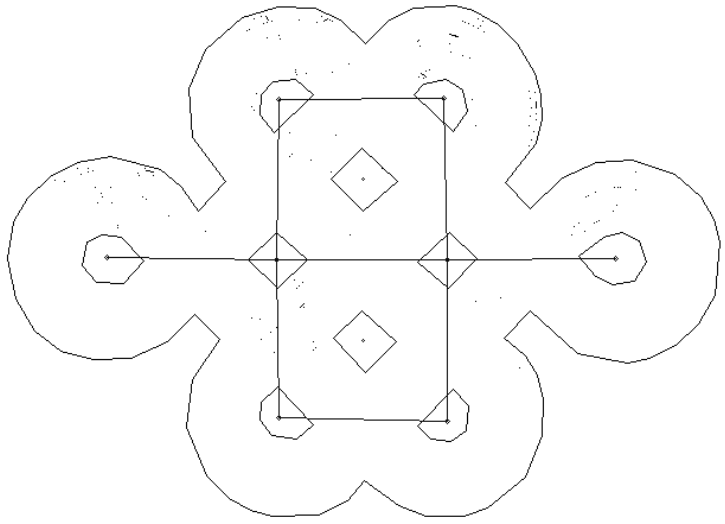


Figure 10.15: An example of finding adjacent contours. The small cycles are the centers of edges of the contours. The short segments incident to the small cycles are rays to find adjacent contours.

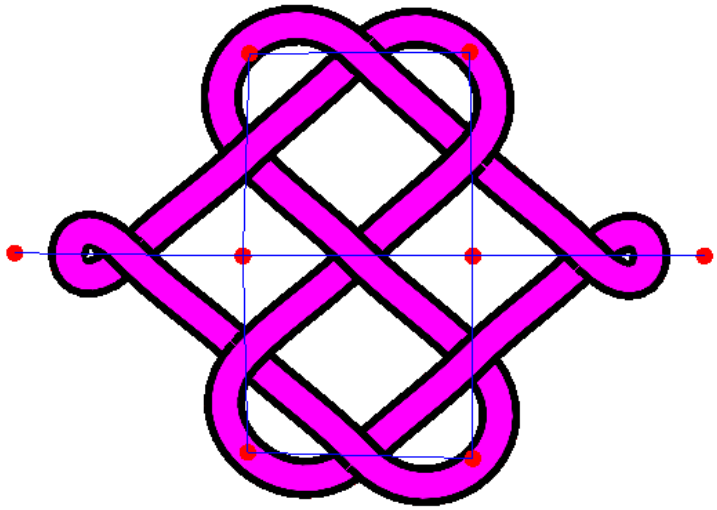
ConstructGraph
INPUT: an array A of contours, which are labeled with V-gap and F-gap.
OUTPUT: a constructed planar graph.

1 **for** each contour p_i bounding a V-gap
1.1 Put a vertex v_i in the geometric center of p_i;
2 **for** each contour p_i bounding a F-gap
2.1 Traverse the edges of contour p_i in clockwise order
2.2 Obtain an order of the adjacent V-gaps by the order of the hitting rays of the edges
3 **for** any two consecutive V-gaps p_j, p_k in the order
3.1 **if** there is no edge between v_i, v_k.
3.2 Add an edge between the vertices v_j, v_k.

Figure 10.16: Algorithm for classify gaps into F-gap and V-gap



(a)



(b)

Figure 10.17: An example of graph construction(a) and its induced weaving(b).

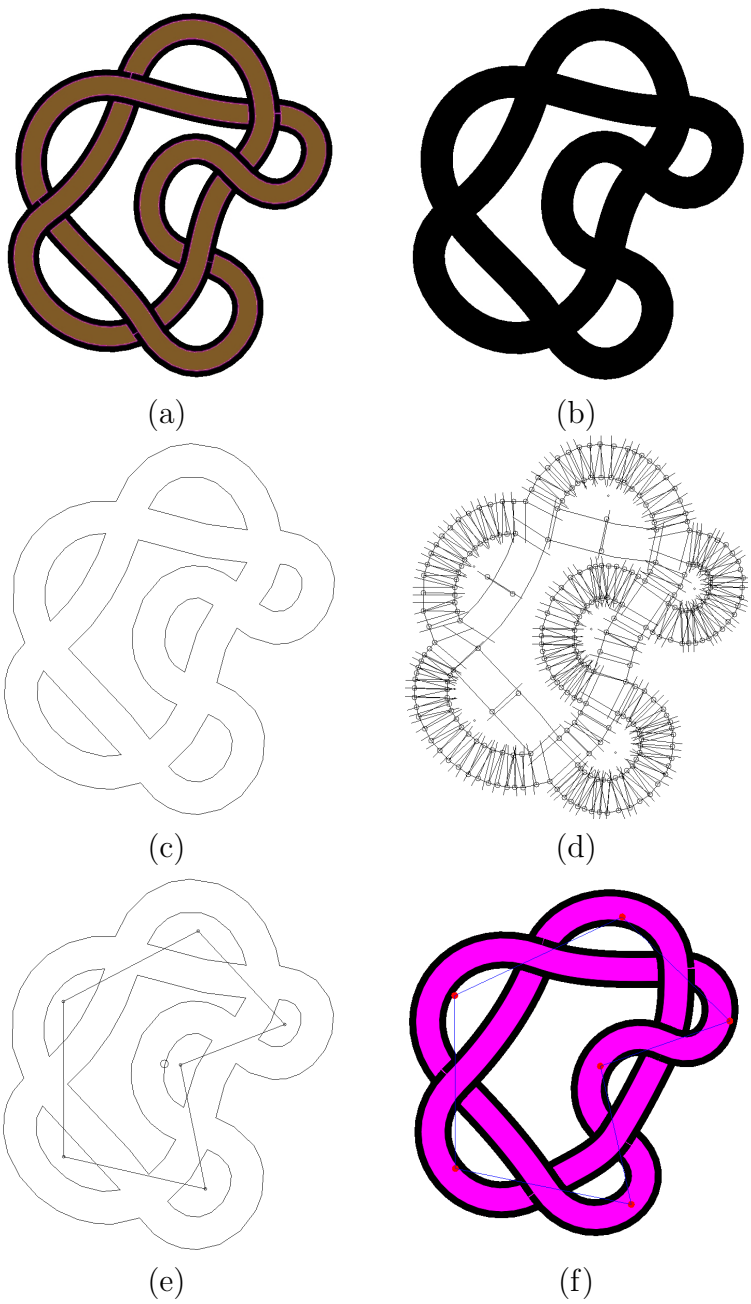


Figure 10.18: An example for weaving reconstruction(1). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving

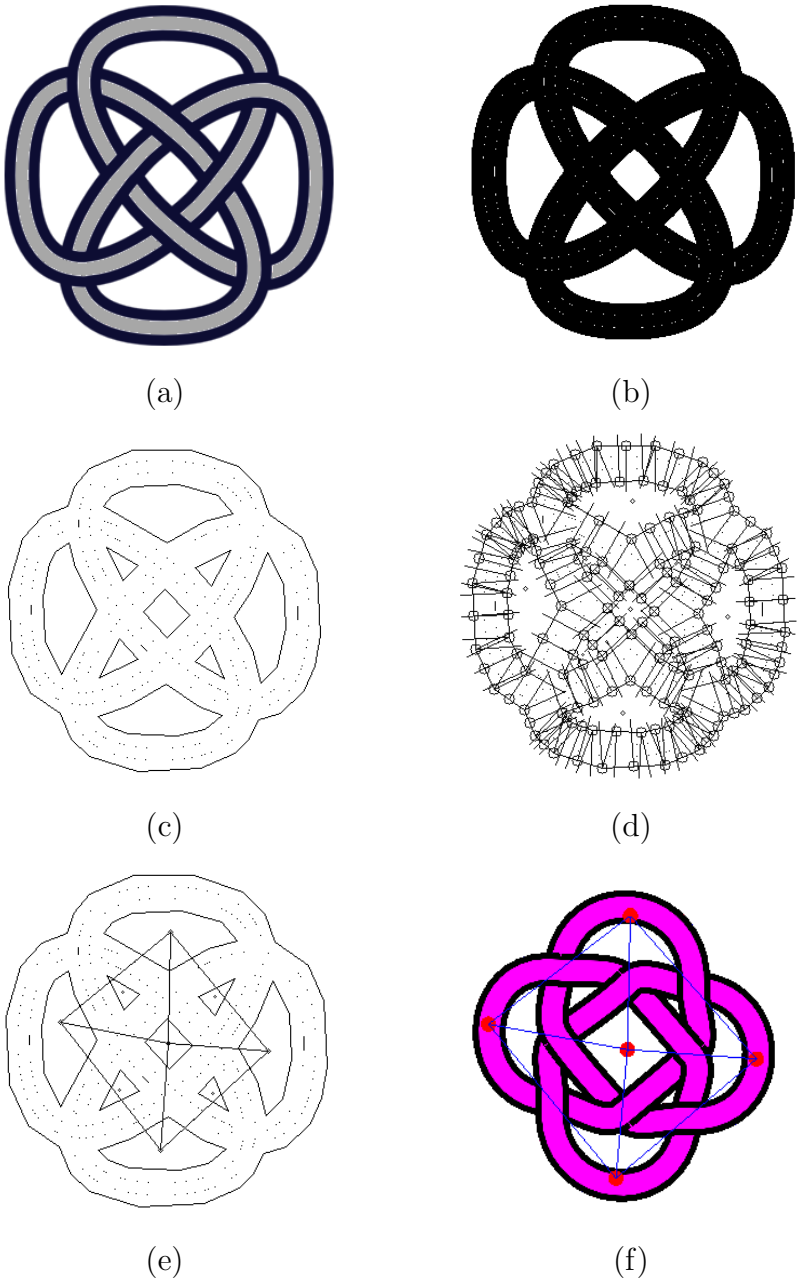


Figure 10.19: An example for weaving reconstruction(2). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving

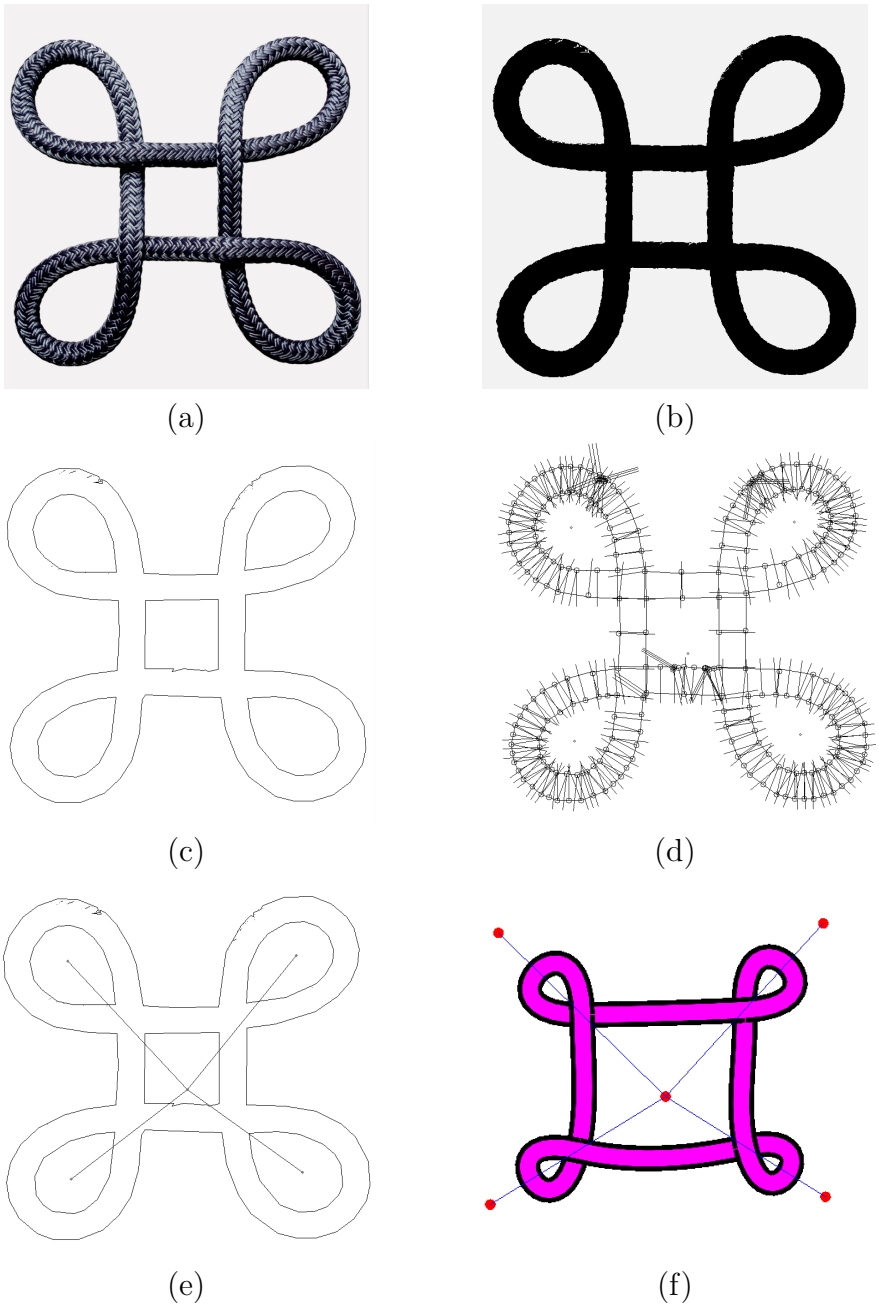


Figure 10.20: An example for weaving reconstruction(3). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving

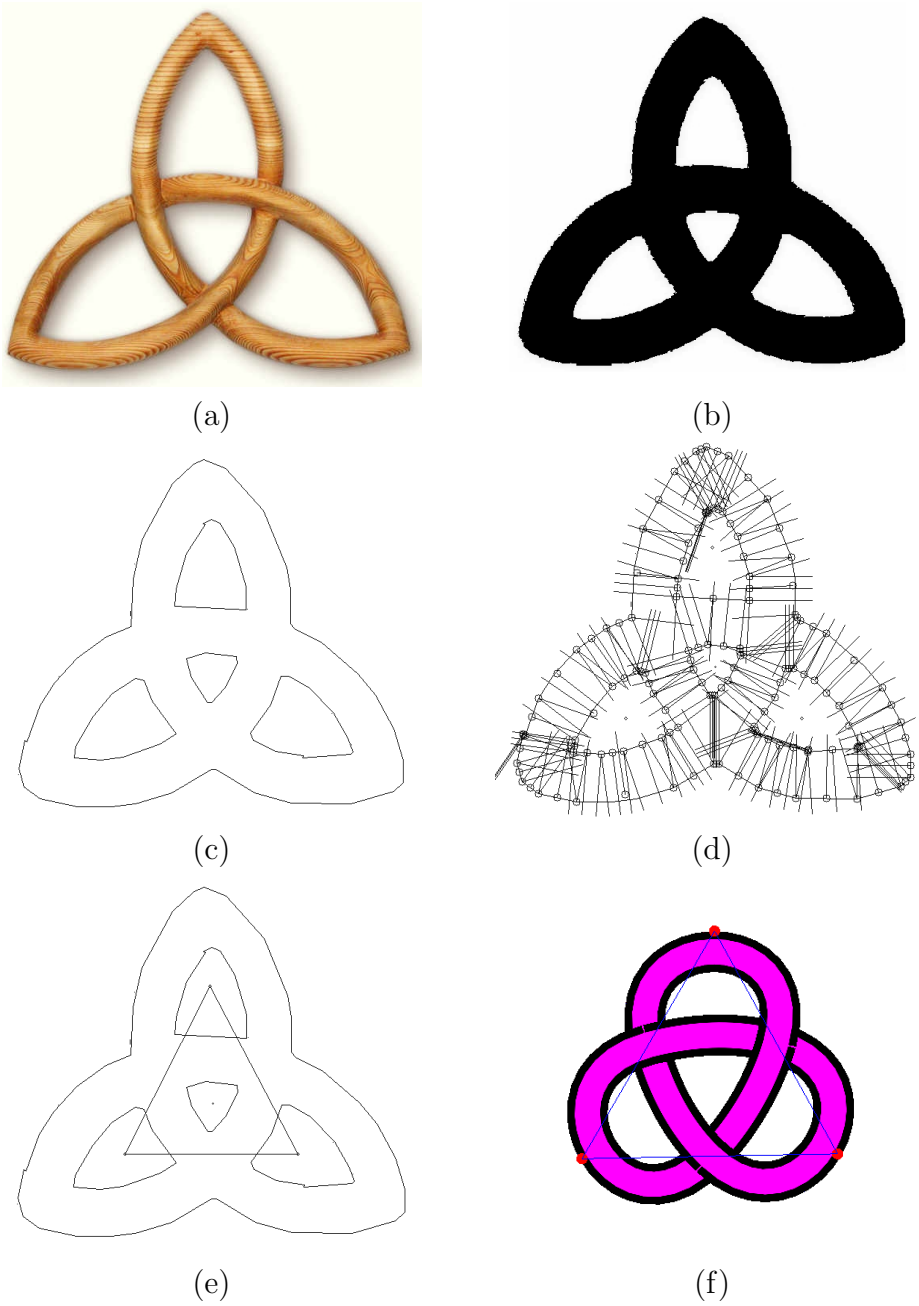


Figure 10.21: An example for Trefoil knot reconstruction(4). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving

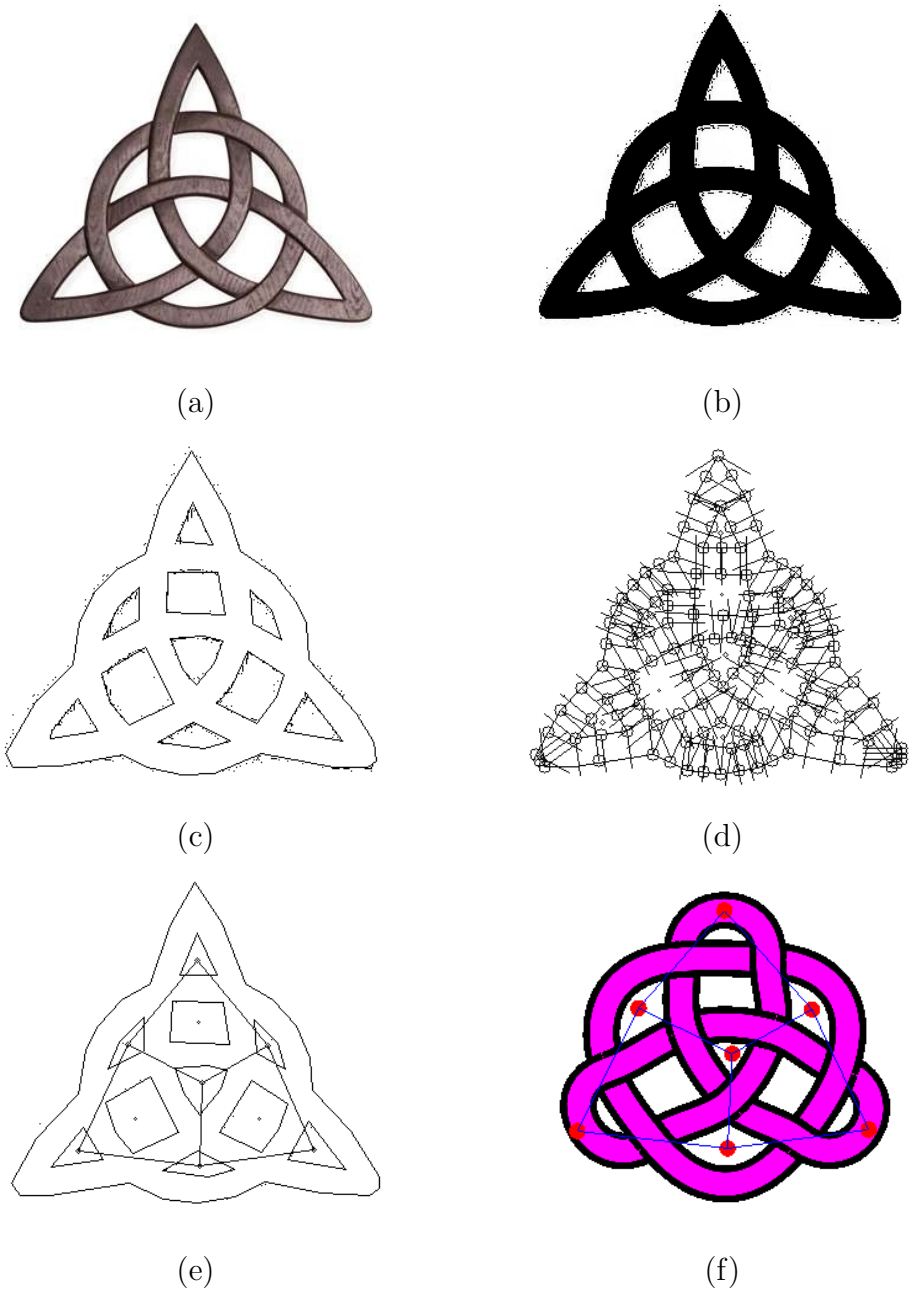


Figure 10.22: An example for weaving reconstruction(5). (a) Original image; (b) Binary image; (c) Detected contours; (d) Adjacent gap detection; (e) Reconstructed image; (f) Reconstructed weaving

11. CONCLUSION

We have presented a framework for studying weaving structures on arbitrary surfaces. This framework is based on an extended version of graph rotation system (ERS), which originated from a fundamental concept in the study of graph imbeddings. We have introduced the concepts of cellular weaving and normal weaving on general surfaces, and we have given necessary and sufficient conditions for an ERS to induce a cellular/normal weaving structure. To handle most weavings in practice, we have extended our research to the imbeddings of disconnected graphs, and based on this we have presented methods to induce any weaving on a plane and any 2-colorable weaving on a torus, while the weavings on the plane have been the focus of previous research in weaving construction.

We have also studied surgery operations on weaving structures and their corresponding operations on ERSs. The two most frequently invoked subdivision operations in computer graphics, Catmull-Clark and Doo-Sabin operations, have the same effects with our doubling operation on induced weavings. The doubling operation does not only provide a way of refining a weaving, but could also transfer a special type of weavings, which may not be twillable, into twillable weavings. We have studied this special type of weavings, and have realized that its highly symmetric structures can be represented by permutation voltage graph with a simple fixed base graph. Inspired by the symmetric structure, we further extended our research into computer graphics. We have identified a type of mesh, called Quad-Pattern Coverable mesh, which could be seamlessly covered by one single periodic pattern. Similarly, its topological structure can be simply represented by Permutation Voltage graph with a bouquet with two self-loops as the base graph.

After the work above, we summarized the properties offered by the weaving con-

struction framework as follows.

1. *Topological completeness:* the system can produce visual woven images on any closed surface in 3-space;

2. *Operational robustness:* all the operations in our graphic system correspond to authentic topological operations, which provide secure control over the result of applying these operations, including certainty that the result is the intended topological object;

3. *Conciseness:* the graph imbedding and edge-twistings used to specify a weaving can be given by any of the extensively studied representations of graph imbeddings;

4. *Flexibility:* many dynamic surgery computer-graphics operations on weavings now can be implemented via their theoretical counterparts on the corresponding graph imbeddings. which have been extensively studied in the graph theory literatures.

In practice, we have implemented a computer-graphic system based on this theoretical framework (see [1]), which is already in widespread use. Moreover, we have presented a software for plane weaving construction in the last chapter. The software has implemented all necessary operations to create a plane weaving with the consideration of geometry issues. Furthermore, the software also accepts high-quality weaving images as input, and could detect and reconstruct the weavings in images. This function could make records and modifications of existing weavings much more conveniently.

Many interesting questions remain unanswered in this line of research.

One question is whether there are weaving structures that cannot be induced by an ERS. By Corollary 4.2.3, a weaving induced by an ERS of a connected graph must be a normal weaving. Therefore, no abnormal weaving can be generated by this method from a connected graph. According to Corollary 5.2.6, any plane weaving and a weaving with 2-colorable sub-weavings on a torus are inducible from ERSs of organized disconnected graphs, i.e., tree structure is used to organize disconnected plane graph and cellular-based structure is used for weavings on torus. We have not found "meaningful" structures to describe abnormal weavings on higher genus surface. As a further research topic, it will be very interesting to investigate the power of this method based on ERSs and their variations.

All the weavings discussed here, normal, abnormal and plane, require the condition of gap 2-colorability. Therefore, the gap 2-colorability seems an important feature of weaving structures inducible by ERSs, in particular for those on higher genus surfaces, which deserve further study.

It would be an interesting topic to explore more surgery operations on a weaving structure, such as to add or delete one strand from a weaving. Based on our weaving construction framework, a strand in a weaving is an edge-walk in the corresponding imbedded graph, where the next edge in a walk may be 0-next edge or 1-next edge at the end vertex. It is not obvious whether the obtained weaving after adding or deleting a strand is still inducible or not. If the obtained weaving is inducible, the structure of the corresponding ERS is not intuitive. The investigation of this topic might cover more properties of the weaving construction framework.

Theorem 6.4.3 shows a rather interesting fact that two simply-twisted ERSs whose pure rotation systems (i.e., the corresponding graph imbeddings) are dual to each other induce essentially the same weaving. In fact, it is not difficult to see that simply-

twisting an edge in the primary imbedding and simply-twisting its corresponding edge in the dual imbedding locally produce essentially the same weaving effect in the induced weavings: they both make a cross across the middle of the edges (note that we can suppose that the primary edge and dual edge share the same "edge middle"). On the one hand, it seems difficult to implement the weaving effect of an untwisted edge in the primary imbedding by operating on the corresponding edge in the dual imbedding. On the other hand, this issue seems closely related to the concept of "partial dual" recently introduced by Chmutov[12]. Therefore, by considering partial dual graphs, it may be possible to further extend Theorem 6.4.3 to larger classes of weaving structures.

Another research direction is to study weaving structures on surfaces with boundaries, which are commonly found in daily life (e.g., baskets and sweaters). A straightforward way to implement this is to simply treat some selected gaps in a weaving as "holes" on the corresponding surface; thus, the weaving can also be regarded as a weaving on the surface with the holes. On the other hand, this subject seems closely related to the classical study on spanning surfaces in knot theory[42]. Note that any compact, connected, and oriented surface with non-empty boundary in Euclidean 3-space is a spanning surface associated to its boundary link[28]. Therefore, a weaving structure on a surface with boundary is a weaving on a spanning surface. This suggests an interesting 2-level construction of weaving structures on surfaces with boundaries: the first level provides a link structure to describe the underlying surface with boundaries, and the second level gives the weaving structure on the resulting surface. Note that Seifert's algorithm for constructing Seifert surfaces[42] has many analogs with the operations on weaving structures that we have discussed in the current paper. Relationship between these two subjects is certainly interesting and deserves further investigation.

In practice, our software to reconstruct plane weavings from images is still at an initial stage which requires high image quality. It would be useful and interesting for artists to reconstruct weavings from more general weaving images in real, i.e., images of Celtic knots. However, this research direction is mainly related to image processing, which is out of the scope of this paper.

REFERENCES

[1] Ergun Akleman, Jianer Chen, YenLin Chen, Qing Xing, and Jonathan L Gross. Cyclic twill-woven objects. *Computers & Graphics*, 35(3):623–631, 2011.

[2] Ergun Akleman, Jianer Chen, and Jonathan L Gross. Paper-strip sculptures. In *Shape Modeling International Conference (SMI), 2010*, pages 236–240. IEEE, 2010.

[3] Ergun Akleman, Jianer Chen, Qing Xing, and Jonathan L Gross. Cyclic plain-weaving on polygonal mesh surfaces with extended graph rotation systems. In *Proc. SIGGRAPH*, volume 9, 2009.

[4] Anni Albers. *Anni Albers: On Weaving*. Wesleyan Univ. Press, Middletown, Conn., 1963.

[5] Dan Archdeacon. The medial graph and voltage-current duality. *Discrete mathematics*, 104(2):111–141, 1992.

[6] George Bain. *Celtic art: the methods of construction*. Courier Dover Publications, New York, NY, 1951.

[7] Iain Bain. *Celtic knotwork*. Constable, London, 1986.

[8] Robert Berger. *The undecidability of the domino problem*, volume 66. American Mathematical Soc., 1966.

[9] Edwin Catmull. A subdivision algorithm for computer display of curved surfaces. Technical report, DTIC Document, 1974.

[10] Edwin Catmull and James Clark. Recursively generated b-spline surfaces on arbitrary topological meshes. *Computer-aided design*, 10(6):350–355, 1978.

[11] Professor Cayley. Desiderata and suggestions: No. 2. the theory of groups: graphical representation. *American Journal of Mathematics*, 1(2):174–176, 1878.

[12] Sergei Chmutov. Generalized duality for graphs on surfaces and the signed bollobás–riordan polynomial. *Journal of Combinatorial Theory, Series B*, 99(3):617–638, 2009.

[13] Michael F Cohen, Jonathan Shade, Stefan Hiller, and Oliver Deussen. *Wang tiles for image and texture generation*, volume 22. ACM, 2003.

[14] Ann M Collier. *A handbook of textiles*. Pergamon Press, Oxford, UK, 1970.

[15] Peter R Cromwell. Celtic knotwork: Mathematical art. *The Mathematical Intelligencer*, 15(1):36–47, 1993.

[16] Daniel Doo and Malcolm Sabin. Behaviour of recursive division surfaces near extraordinary points. *Computer-Aided Design*, 10(6):356–360, 1978.

[17] Jack Edmonds. A combinatorial representation of polyhedral surfaces. *Notices of the American Mathematical Society*, 7, 1960.

[18] Alexei A Efros and William T Freeman. Image quilting for texture synthesis and transfer. In *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*, pages 341–346. ACM, 2001.

[19] Shanzhen Gao, Michal Sramka, and Zhonghua Tan. Graphs on surfaces. *Congressus Numerantium*, 177:133, 2005.

[20] Jonathan L Gross. Every connected regular graph of even degree is a schreier coset graph. *Journal of Combinatorial Theory, Series B*, 22(3):227–232, 1977.

[21] Jonathan L Gross and Thomas W Tucker. *Topological graph theory*. Courier Dover Publications, New York, NY, 1987.

[22] Jonathan L Gross and Thomas W Tucker. A celtic framework for knots and links. *Discrete & Computational Geometry*, 46(1):86–99, 2011.

[23] Branko Grünbaum and Geoffrey C Shephard. Satins and twills: An introduction to the geometry of fabrics. *Mathematics Magazine*, 53(3):139–161, 1980.

[24] Branko Grunbaum and Geoffrey C Shephard. *Tilings and patterns*. W.H. Freeman & Company, New York, NY, 1986.

[25] Lothar Heffter. Über das problem der nachbargebiete. *Mathematische Annalen*, 38(4):477–508, 1891.

[26] Shiyu Hu, Qing Xing, Ergun Akleman, Jianer Chen, and Jonathan Gross. Pattern mapping with quad-pattern-coverable quad-meshes. *Computers & Graphics*, 2012.

[27] Matthew Kaplan and Elaine Cohen. Computer generated celtic design. In *Proceedings of the 14th Eurographics workshop on Rendering*, pages 9–19. Eurographics Association, 2003.

[28] Louis H Kauffman. *Formal knot theory*. Courier Dover Publications, Mineola, NY, 2006.

[29] Yuanyuan Li, Eugene Zhang, Yoshihiro Kobayashi, and Peter Wonka. Editing operations for irregular vertices in triangle meshes. In *ACM Transactions on Graphics (TOG)*, volume 29, page 153. ACM, 2010.

[30] Samuel J Lomonaco and Louis H Kauffman. Quantum knots and mosaics. *Quantum Information Processing*, 7(2-3):85–115, 2008.

[31] Charles Loop. *Smooth subdivision surfaces based on triangles*. Master's thesis, Department of Mathematics, University of Utah, August, 1987.

[32] Vassily Manturov. *Knot theory*. CRC Press, Boca Raton, Fla., 2004.

[33] William S Massey. *Algebraic topology: an introduction*. Springer-Verlag, New York, NY, 1987.

[34] Aidan Meehan. *Celtic design: a beginner's manual*. Thames and Hudson, New York, NY, 1991.

[35] Aidan Meehan. *Knotwork: the secret method of the scribes*. Thames and Hudson, New York, NY, 1991.

[36] Aidan Meehan. *Celtic Design: Illuminated Letters*, volume 2. Thames and Hudson, New York, NY, 1992.

[37] Aidan Meehan. *Celtic Borders*. Thames and Hudson, New York, NY, 1999.

[38] Christian Mercat. Les entrelacs des enluminures celtes. *Pour la Science*, 1997.

[39] Harry Nisbet. *Grammar of textile design*. Van Nostrand, New York, NY, 1927.

[40] Gustaf Hermann Oelsner. *A handbook of weaves*. Macmillan Company, New York, NY, 1915.

[41] Chi-Han Peng, Eugene Zhang, Yoshihiro Kobayashi, and Peter Wonka. Connectivity editing for quadrilateral meshes. In *ACM Transactions on Graphics (TOG)*, volume 30, page 141. ACM, 2011.

[42] Herbert Seifert. Über das geschlecht von knoten. *Mathematische Annalen*, 110(1):571–592, 1935.

[43] Ariel Shamir. A survey on mesh segmentation techniques. *Computer Graphics Forum*, 27(6):1539–1556, 2008.

[44] Alexander Soifer. *The mathematical coloring book: Mathematics of coloring and the colorful life of its creators*. Springer, New York, NY, 2009.

[45] Cyril Soler, Marie-Paule Cani, and Alexis Angelidis. Hierarchical pattern mapping. In *ACM Transactions on Graphics (TOG)*, volume 21, pages 673–680. ACM, 2002.

[46] J. Stam. Hierarchical pattern mapping. Aperiodic Texture Mapping. Technical Report R046, European Research Consortium for Informatics and Mathematics (ERCIM), 1997.

[47] John Henry Strong. *Foundations of fabric structure*. National Trade Press, London, 1953.

[48] James Trilling. *The language of ornament*. Thames & Hudson, New York, NY, 2001.

[49] Wikipedia. Celtic knot. *http://en.wikipedia.org/wiki/Celtic_knot*.

[50] I-loo Wong. Games, logic and computers. *Scientific American*, 213:98–106, 1965.

[51] Cem Yuksel, Jonathan M Kaldor, Doug L James, and Steve Marschner. Stitch meshes for modeling knitted clothing with yarn-level detail. *ACM Transactions on Graphics (TOG)*, 31(4):37, 2012.